

# Шпаргалка. Операции командной строки и Git

## Основные операции командной строки

- `pwd` — вывести на экран путь до текущей папки.
- `ls` — показать содержимое текущей папки;
  - `ls -la` — показать содержимое, включая скрытые файлы.
- `cd path` — перейти в папку `path` ;
  - `cd ..` — перейти на одну директорию вверх;
  - `cd ~` — перейти в домашнюю директорию текущего пользователя..
- `mkdir dirname` — создать директорию с именем `dirname` .
- `touch filename` — создать пустой файл с названием `filename` .
- `cp path/to/file/filename new/path/directory` — скопировать файл с названием `filename` из папки `path/to/file/` в папку `new/path/directory` .
- `mv path/to/file/filename new/path/directory` — переместить файл с названием `filename` из папки `path/to/file/` в папку `new/path/directory` .
- `rm filename` — удалить файл с названием `filename` ;
  - `rm -r directory` — удалить рекурсивно (на это указывает `-r` ) все файлы из директории `directory` , а затем саму директорию.

## Основные операции Git

- `git clone repo_https` — клонировать репозиторий с указанным `repo_https` на GitHub. Сделать это нужно только один раз, при первоначальной настройке проекта. Чтобы узнать `repo_https` вашего репозитория, перейдите на его главную страницу, в правом верхнем углу кликните на **Code**, во вкладке **Local** скопируйте строку **HTTPS**.
- `git pull` — скачать последнюю версию проекта из Github.

- `git add file1 file2` — добавить изменения в файлах `file1` и `file2` в проект.
- `git status` — посмотреть изменения в файлах.
- `git commit -m 'commit message'` — зафиксировать новую версию проекта с кратким комментарием об изменении `commit message`.
- `git push` — отправить новую версию проекта на GitHub.