git add - проиндексировать

git commit - добавить в локальный репозиторий

git push - добавить на сервер

git status

git status -s

git help <команда>

git <команда> --help

man git-<команда>

git config --list - проверка настроек

git clone - клонирование существующего репозитория

cd C:/Users/user/my\_project и git init - создание репозитория в существующей директории

git show - показывает информацию

git log - просмотр коммитов

git log --pretty=gormat:”%h - %s”

git log --pretty=oneline

git tag - просмотр тегов

git log --oneline --decorate - просмотр, куда указывают указатели веток

git log <nameBranch> - просмотр логов в заданной ветке

git log --all - просмотр логов во всех ветках

git log --oneline --decorate --graph --all

git tag - просмотр тегов

git merge <nameBranch> - слияние ветки в которой вы находитесь с nameBranch

git branch <nameBranch> - создание новой ветки

git checkout <nameBranch> - переключение на существующую ветку

git checkout -b <newbranchname> - создание новой ветки и переключение на нее

git branch -d <nameBranch> удаление ветки

git branch - просмотр всех веток

git branch -v - последний коммит на каждой из веток

--merged и --no-merged - фильтруют вывода только тех веток, которые слиты или ещё не слиты в текущую ветку

git push <remote> <branch> - отправка ветки на сервер

git tag -lw - легковесный тег

git tag -a - аннотированный тег

git tag -a <tagName> <commitNumber> - для добавления тега к указанному коммиту (номер первые 7 цифр)

git push origin <tagName> - для отправки тега на сервер

git push origin --tags - для отправки всех тегов

git tag -d <tagName> - удаление тега из локального репозитория

git push <remote> :refs/tags/<tagname> - удаление тега с сервера

git push origin --delete <tagname> - удаление тега с сервера

git add -h

usage: git add [<options>] [--] <pathspec>...

-n, --dry-run dry run

-v, --verbose be verbose

-i, --interactive interactive picking

-p, --patch select hunks interactively

-e, --edit edit current diff and apply

-f, --force allow adding otherwise ignored files

-u, --update update tracked files

--renormalize renormalize EOL of tracked files (implies -u)

-N, --intent-to-add record only the fact that the path will be added later

-A, --all add changes from all tracked and untracked files

--ignore-removal ignore paths removed in the working tree (same as --no-all)

--refresh don't add, only refresh the index

--ignore-errors just skip files which cannot be added because of errors

--ignore-missing check if - even missing - files are ignored in dry run

--chmod (+|-)x override the executable bit of the listed files

Псевдонимы:

$ git config --global alias.co checkout

$ git config --global alias.br branch

$ git config --global alias.ci commit

$ git config --global alias.st status

$ git config --global alias.unstage 'reset HEAD --'

$ git config --global alias.last 'log -1 HEAD'

$ git config --global alias.visual "!gitk"

Ошибки слияния: перейти в файл, там должны быть пометки и выбрать нужное, сделать add для каждой ветки, Git поймет, что конфликт устранен и сам сделает слияние.