

Помощь	Изменения	Отслеживание	Работа с ветками	Синхронизация
Помощь по Git в терминале \$ git --help	сделать указанный файл готовым для коммита. \$ git add [file]	Полный список изменений файлов, ожидающих коммита. \$ git status	список всех локальных веток в текущей директории. \$ git branch	запустить текущую ветку в удаленную ветку. \$ git push
Создание репозиториев	сделать все измененные файлы готовыми для коммита. \$ git add .	Краткий вид изменений \$ git status -s	Показать все локальные и удалённые(remotes) ветки \$ git branch -av	запустить ветку в указанный репозиторий и удаленную ветку. \$ git push [remote] [branch]
С нуля. Создать новый локальный репозиторий с заданным именем. \$ git init [project-name]	добавить только файлы, соответствующие указанному выражению. \$ git add "*.txt"	Показать изменения в файлах, которые еще не были добавлены в индекс коммита (staged) \$ git diff	создать новую ветку \$ git branch [branch-name]	если удаленная ветка не установлена как отслеживаемая, то сделать ее такой. \$ git push -u origin main
Клонирова имеющийся. загрузить проект и его полную историю изменений. \$ git clone [url]	записать изменения в репозиторий. для написания сообщения откроется назначенный редактор. \$ git commit	показать что было добавленно в индекс с помощью git add, но еще не было закоммиченно. \$ git diff --staged	переключиться на указанную ветку и обновить рабочую директорию \$ git checkout [branch-name]	указать конкретную удаленную ветку для слияния. \$ git pull [remote] [branch]
Конфигурация	git commit -m "[descriptive message]" — записать изменения с заданным сообщением. \$ git commit -m "[descriptive message]"	список изменения текущей ветки \$ git log	создать и переключиться на новую ветку \$ git checkout -b	соединить изменения в текущей ветке с изменениями из заданной. \$ git merge [branch]
установить имя, которое будет прикрепляться к коммиту. \$ git config --global user.name "[name]"	git commit --amend — добавить изменения к последнему коммиту. \$ git commit --amend	посмотреть на файл в другой ветке, не переключаясь на неё. \$ git show [branch]:[file]	переименовать ветку \$ git branch -m	посмотреть список доступных удаленных репозиториев. \$ git remote
установить email, который будет прикрепляться к коммиту. \$ git config --global user.email "[email address]"	убрать изменения из индекса коммита, сами изменения останутся. \$ git reset	показать что изменилось с последнего коммита. \$ git diff HEAD	удалить заданную ветку. \$ git branch -d [branch]	посмотреть детальный список доступных удаленных репозиториев. \$ git remote -v
включить полезную подсветку командной строки. \$ git config --global color.ui auto	отменить все коммиты после указанного коммита, изменения будут сохранены локально. \$ git reset [commit/tag]	положить во временное хранилище все отслеживаемые файлы. \$ git stash	вернуть файл в первоначальное состояние если он еще не был добавлен в индекс коммита. \$ git checkout [filename]	добавить новый удаленный репозиторий. \$ git remote add [remote] [url]
обновлять удаленную ветку с таким же именем, что и локальная, при пуше изменений (если не указано иного). \$ git config --global push.default current	принудительно вернуться к указанному коммиту, не сохраняя историю и изменения. \$ git reset --hard [commit]	восстановить последние файлы, положенные во временное хранилище. \$ git stash pop	удалить файл из рабочей директории и добавить в индекс информацию об удалении. \$ git rm [file]	Объединение изменений через rebase локальной и удалённой веток \$ git pull --rebase origin feature
установить редактор для редактирования сообщений коммита. \$ git config --global core.editor [editor]	Откатить всё до последней фиксации \$ git reset --hard	список всех сохраненных изменений во временном хранилище. \$ git stash list	удалить файл из репозитория, но сохранить его локально. \$ git rm --cached [file]	загрузить всю историю с заданного удаленного репозитория. \$ git fetch [remote]
		удалить последние файлы, положенные во временное хранилище. \$ git stash drop	изменить имя файла и добавить в индекс коммита. \$ git mv [file-original] [file-renamed]	Сделано на www.andron13.de