

Первоначальная настройка	Шаблон команды	Примеры
Устанавливает имя, которое будет отображаться в поле автора у выполняемых вами коммитов	git config --global user.name "[имя]"	git config --global user.name "Yandex Student"
Задаёт адрес электронной почты, которая будет использоваться для всех коммитов	git config --global user.email "[адрес электронной почты]"	<a href="mailto:student@yandex.ru">git config --global user.email "student@yandex.ru"</a>
Позволяет управлять подключениями к репозиториям	git remote	git remote set-url origin ssh://git@github.com/Yandex-Praktikum-DevOps/devops-basic.git
Базовые операции		
Создание репозитория в текущей директории	git init [название проекта]	git init my_cool_project
Скачать удаленный репозиторий	git clone [url-адрес]	git clone https://github.com/GuidoBartoli/sherlok.git; git clone git@github.com:GuidoBartoli/sherlok.git;
Синхронизация с удаленным репозиторием		
Скачивает историю из удалённого репозитория	git fetch [удалённый репозиторий]	
Загружает все изменения локальной ветки в удалённый репозиторий	git push [удалённый репозиторий] [ветка]	
Загружает историю из удалённого репозитория и объединяет её с локальной. pull = fetch + merge	git pull	git pull
Внесение изменений		
Перечисляет все новые или изменённые файлы, которые нуждаются в фиксации в индексе	git status	git status
Добавляем файлы в Index	git add [файл]	git add index.html; git add . ; git add --all ;
Удаляет файл из рабочей директории и индексирует его удаление	git rm [файл]	git rm index.html
Убирает файл из контроля версий, но физически оставляет его на своём месте	git rm --cached [файл]	git rm --cached index.html
Перемещает или переименовывает файл, сразу индексируя его для последующего коммита	git mv [оригинальный файл] [новое имя]	git mv index.html index.txt
Показывает различия по внесённым изменениям в ещё не проиндексированных файлах	git diff	git diff
Показывает различия между проиндексированной и последней зафиксированной версиями файлов	git diff --staged	git diff --staged
История изменений конкретного файла	git log --follow [файл]	git log --follow index.txt
Коммиты		
Фиксирует проиндексированные изменения и сохраняет их в историю версий	git commit -m "[описание коммита]"	git commit -m "Added new feature."
Изменение сообщения последнего коммита	git commit --amend -m "новое описание к коммиту"	git commit --amend -m "Added new feature. Task in Jira 3247"
Просмотр изменений(коммитов) проекта	git log	git log
Выводит информацию и показывает изменения в выбранном коммите	git show [номер коммита]	git show 73b64290a107f6325ab88c7950637a4ce44c42a6
Отменяет все коммиты после заданного, оставляя все изменения в рабочей директории	git reset [номер коммита]	git reset 73b64290a107f6325ab88c7950637a4ce44c42a6
Сбрасывает всю историю вместе с состоянием рабочей директории до указанного коммита.	git reset --hard [коммит]	git reset --hard 73b64290a107f6325ab88c7950637a4ce44c42a6
Просмотр изменений (коммитов) проекта виде графа	git log --graph --oneline --stat	git log --graph --oneline --stat

<b>Ветвления</b>			
Список веток коммитов	git branch	git branch	
Создать новую ветку	git branch [имя ветки]	git branch feature1	
Переключиться на выбранную ветку	git checkout [имя ветки]	git checkout feature1	
Удалить выбранную ветку	git branch -d [имя ветки]	git branch -d feature1	
Переименование ветки	git branch -m [старое_название_ветки] [новое_название_ветки]	git branch -m feature1 SuperButtonFeature	
Слияние веток (вносит изменения указанной ветки в текущую ветку)	git merge [имя ветки]	git merge hotfix	