

Внесение изменений

Просмотр изменений и создание коммитов (фиксация изменений)
<div><div>\$ git status</div></div>
Перечисляет все новые или изменённые файлы, которые нуждаются в фиксации
<div><div>\$ git diff</div></div>
Показывает различия по внесённым изменениям в ещё не проиндексированных файлах

Первоначальная настройка

Настройка информации о пользователе для всех локальных репозиториев
<div><div>\$ git config --global user.name "[имя]"</div></div>
Устанавливает имя, которое будет отображаться в поле автора у выполняемых вами коммитов
<div><div>\$ git config --global user.email "[адрес электронной почты]"</div></div>
Устанавливает адрес электронной почты, который будет отображаться в информации о выполняемых вами коммитах

Создание репозитория

Создание нового репозитория или получение его по существующему URL-адресу
<div><div>\$ git init [название проекта]</div></div>
Создаёт новый локальный репозиторий с заданным именем
<div><div>\$ git clone [url-адрес]</div></div>
Скачивает репозиторий вместе со всей его историей изменений

Операции с файлами

Перемещение и удаление версий файлов репозитория
<div><div>\$ git rm [файл]</div></div>
Удаляет конкретный файл из рабочей директории и индексирует его удаление
<div><div>\$ git rm --cached [файл]</div></div>
Убирает конкретный файл из контроля версий, но физически оставляет его на своём месте
<div><div>\$ git mv [оригинальный файл] [новое имя]</div></div>
Перемещает и переименовывает указанный файл, сразу индексируя его для последующего коммита

Игнорирование некоторых файлов

Исключение временных и вторичных файлов и директорий
<div><div><pre>*.log build/ temp-*</pre></div></div>
Git будет игнорировать файлы и директории, перечисленные в файле <code>.gitignore</code> с помощью wildcard синтаксиса
<div><div>\$ git ls-files --others --ignored --exclude-standard</div></div>
Список всех игнорируемых файлов в текущем проекте

Сохранение фрагментов

Сохранение и восстановление незавершённых изменений
<div><div>\$ git stash</div></div>
Временно сохраняет все незафиксированные изменения отслеживаемых файлов
<div><div>\$ git stash pop</div></div>
Восстанавливает состояние ранее сохранённых версий файлов
<div><div>\$ git stash list</div></div>
Выводит список всех временных сохранений
<div><div>\$ git stash drop</div></div>
Сбрасывает последние временно сохранённые изменения

<div><div>\$ git add [файл]</div></div>
Индексирует указанный файл для последующего коммита
<div><div>\$ git diff --staged</div></div>
Показывает различия между проиндексированной и последней зафиксированной версиями файлов
<div><div>\$ git reset [файл]</div></div>
Отменяет индексацию указанного файла, при этом сохраняет его содержимое
<div><div>\$ git commit -m "[сообщение с описанием]"</div></div>
Фиксирует проиндексированные изменения и сохраняет их в историю версий

Коллективная работа

Именованные серии коммитов и соединение результатов работы
<div><div>\$ git branch</div></div>
Список именованных веток коммитов с указанием выбранной ветки
<div><div>\$ git branch [имя ветки]</div></div>
Создаёт новую ветку
<div><div>\$ git checkout [имя ветки]</div></div>
Переключается на выбранную ветку и обновляет рабочую директорию до её состояния
<div><div>\$ git merge [имя ветки]</div></div>
Вносит изменения указанной ветки в текущую ветку
<div><div>\$ git branch -d [имя ветки]</div></div>
Удаляет выбранную ветку

Просмотр истории

Просмотр и изучение истории изменений файлов проекта
<div><div>\$ git log</div></div>
История коммитов для текущей ветки
<div><div>\$ git log --follow [файл]</div></div>
История изменений конкретного файла, включая его переименование
<div><div>\$ git diff [первая ветка]...[вторая ветка]</div></div>
Показывает разницу между содержанием коммитов двух веток
<div><div>\$ git show [коммит]</div></div>
Выводит информацию и показывает изменения в выбранном коммите

Откат коммитов

Удаление ошибок и корректировка созданной истории
<div><div>\$ git reset [коммит]</div></div>
Отменяет все коммиты после заданного, оставляя все изменения в рабочей директории
<div><div>\$ git reset --hard [коммит]</div></div>
Сбрасывает всю историю вместе с состоянием рабочей директории до указанного коммита.

Синхронизация с удалённым репозиторием

Регистрация удалённого репозитория и обмен изменениями
<div><div>\$ git fetch [удалённый репозиторий]</div></div>
Скачивает всю историю из удалённого репозитория
<div><div>\$ git merge [удалённый репозиторий]/[ветка]</div></div>
Вносит изменения из ветки удалённого репозитория в текущую ветку локального репозитория
<div><div>\$ git push [удалённый репозиторий] [ветка]</div></div>
Загружает все изменения локальной ветки в удалённый репозиторий
<div><div>\$ git pull</div></div>
Загружает историю из удалённого репозитория и объединяет её с локальной. pull = fetch + merge