

для чего нужен гит

Git является распределенной системой для управления версиями разрабатываемых файлов. Создана она была в 2005 году автором ОС Linux. Эта система осуществляет синхронизацию работы с сайтом, а также сохраняет и обновляет изменения в файлах

как пользоватся и основные команды

Установка в операционной системе Windows Если у пользователя нет Linux, а хочется использовать Git, Windows также поддерживает эту систему. И установить ее очень просто. Существует проект msysGit, процедура установки которого является одной из самых простых. Необходимо просто загрузить файл инсталлятора, который можно найти на странице проекта в GitHub, а затем запустить его. По окончании установки на компьютере будет две версии - графическая и консольная.

Первоначальная настройка Git После того как система контроля установлена на компьютер, нужно выполнить кое-какие действия для настройки среды под пользователя. Делается это единожды. При обновлении все настройки сохраняются. Их можно будет поменять в любой момент. Git включает в себя утилиту git config, позволяющую делать настройки и контролировать работу системы, а также внешний вид. Данные параметры могут сохраняться в трех местах: В файле, содержащем значения, которые являются общими для всех пользователей и репозиториев. В файле, содержащем настройки определенного пользователя. В конфигурационном файле, находящемся в текущем репозитории. Такие параметры действуют только для него. Пользовательское имя

В первую очередь после установки необходимо указать имя пользователя, а также электронную почту. Это очень важно, так как каждый коммит (сохранение состояния) содержит эти данные. Они включаются во все передаваемые коммиты и не могут быть изменены впоследствии. Если указать опцию –global, такие настройки нужно будет сделать один раз. Выбор текстового редактора После указания имени нужно выбрать редактор, который будет необходим при наборе сообщений в Git. По умолчанию будет использоваться стандартный редактор операционной системы. Если пользователь захочет использовать другой, нужно прописать это в настройках конфигурационного файла в строке core.editor. Проверка параметров Чтобы знать основы Git, необходимо уметь проверять используемые настройки. Для этого применяется команда git config –list. Она выводит все доступные параметры, которые сможет найти. Некоторые имена настроек могут присутствовать в списке несколько раз. Это происходит из-за того, что Git считывает один ключ из различных файлов. В такой ситуации для каждого ключа используется последнее значение. Есть возможность проверять значения определенных ключей, вписав в команду вместо «--list» - «{ключ}». Как создать репозиторий Достичь этой цели можно двумя способами. Первый заключается в импорте в систему существующего каталога или проекта. Второй – это клонирование с сервера уже существующего репозитория. Создание в данном каталоге Если пользователь решает начать использование Git для уже имеющегося проекта, он должен перейти в каталог и инициализировать систему. Для этого нужна команда git init. Она создает в каталоге подкаталог, где будут находиться все нужные файлы. На данном этапе еще не устанавливается версионный контроль над проектом. Для добавления файлов под контроль нужно их проиндексировать и сделать первую фиксацию изменений. Клонирование репозитория Для получения копии уже существующего репозитория нужна команда git clone. С ее помощью Git получит копию почти всех данных с сервера. Это касается всех версий каждого файла. Очень удобная возможность, так как в случае выхода из строя сервера программист сможет использовать клон на любом клиенте для возврата сервера в то состояние, в каком он был при клонировании. Это похоже на точку восстановления. Удаление файла в Git Удалить из системы любой файл можно, если исключить его из индекса, то есть из отслеживаемых файлов. Для этого нужна команда git rm. Она также убирает файл из рабочего каталога пользователя. Затем нужно выполнить коммит. После него файл попросту исчезнет и отслеживаться больше не будет. Если же он изменен и уже проиндексирован, то применяют принудительное удаление с параметром -f. Такой способ предотвратит удаление тех данных, которые еще не записались в снимок состояния и которые нет возможности восстановить из системы. Отмена изменений В любой момент может появиться необходимость в отмене какого-либо действия. Если пользователь выполнил коммит рано, забыв внести некоторые файлы, то можно перевыполнить его, используя опцию --amend. Такая команда использует для коммита индекса. Если после выполнения последнего сохранения не производилось никаких изменений, то проект будет в таком же состоянии, и появится редактор для комментариев, где пользователь сможет отредактировать все, что нужно. Нужно помнить, что не каждую операцию отмены можно будет отменить. Иногда можно безвозвратно удалить необходимые данные. Следует быть внимательными.