Создание репозитория

№ урока: 2 **Курс:** Основы использования Git

Средства обучения: Git console, Notepad++, Any GUI

Обзор, цель и назначение урока

Научиться инициализировать репозитории, создавать коммиты, просматривать историю коммитов. Рассмотреть альтернативные варианты работы с Git через пользовательский интерфейс.

Изучив материал данного занятия, учащийся сможет:

- Работать с консолью Git.
- Создать репозиторий для проекта локально.
- Добавлять изменения и фиксировать изменения в репозитории.
- Просматривать и навигировать по истории коммитов.
- Использовать команду git help для получения справки.
- Использовать приложения с пользовательским интерфейсом для работы с Git.

Содержание урока

1. Инициализация репозитория
2. Создание коммитов
3. История изменений
4. Приложения с пользовательским интерфейсом для работы с Git

Резюме

- Для создания репозитория git существует два подхода. Первый – инициализировать его локально и отправить на удаленный сервер, второй – инициализировать на сервере и клонировать. Когда репозиторий создан, стандартный рабочий процесс выглядит так:
  1. Вы делаете изменения в файлах в своём рабочем каталоге.
  2. Подготавливаете файлы, добавляя их в область файлов готовых к коммиту (далее будем называть эту область индексом).
  3. Делаете коммит, который помещает файлы из индекса в каталог Git на хранение.
- Каждый **коммит** состоит из изменений файлов, сообщения к коммиту, где обычно указывается кратко какие изменения сделаны, даты и времени, автора, хеша и ветки к которой относится коммит. **Хеш** это строка из 40 шестнадцатеричных символов, вычисляемая на основе содержимого файла или структуры каталога, он уникальный для каждого коммита. Работая с Git, вы будете встречать такие хеши часто, так как в своей базе данных Git сохраняет всё не по именам файлов, а по хешам.
- **git init** – создает репозиторий в текущей директории (создает новую поддиректорию с именем .git, содержащую все необходимые файлы репозитория).
- **git add** – добавляет указанные файлы под версионный контроль. Другими словами, добавляет файлы в индекс для последующего коммита.
- **git commit** – используется для фиксации изменений в репозитории.
- **git diff** – отображение изменений, которые не были фиксированы выполнением коммита.
- **git status** – используется для определения того, какие файлы в каком состоянии находятся.

CyberBionic systematics

CyberBionic Systematics ® 2020
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- **git log** – используется для вывода истории коммитов. Команда выведет последовательность коммитов с метаданными. Чтобы увидеть изменения в каждом коммите при выводе истории использовать команду **git log -p**.
- **git checkout <commit_hash>** - переключение на указанный коммит в истории. При таком переключении состояние репозитория становится - **detached HEAD**, в этом состоянии можно только смотреть файлы, для того чтобы вносить какие-то изменения нужно чтобы HEAD указывал на ветку. (Ветвления будут рассматриваться на следующих занятиях).
- **git help** – вывод справки, **git help <command>** - ввод доступных опций использования команды.

Закрепление материала

- Что нужно сделать для создания локального репозитория?
- Где Git хранит данные о репозитории?
- Можно ли объединить команды добавления в индекс и коммита в одну?
- Где можно найти хеш коммита, если необходимо на него переключится?

Дополнительное задание

Задание
Выберете любое приложение с пользовательским интерфейсом для работы с Git и подключите или откройте в нем созданный репозиторий. Посмотрите историю изменений. Выполните переключение на один из коммитов из истории.

Самостоятельная деятельность учащегося

Задание 1
Создайте локальный репозиторий. Выполните несколько коммитов. Выведите историю коммитов с изменениями в каждом коммите.

Задание 2
Выполните переключение на один из коммитов из истории.

Рекомендуемые ресурсы

Официальная документация Git
https://git-scm.com/docs

Книга Pro Git, Scott Chacon, Ben Straub
https://git-scm.com/book/ru/v2

CyberBionic Systematics ® 2020
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com