

ВВЕДЕНИЕ В СИСТЕМУ КОНТРОЛЯ ВЕРСИЙ

GIT

это инструмент для отслеживания изменений
в коде и управления версиями файлов

Основная цель

обеспечить возможность работы над проектом несколькими разработчикам одновременно и сохранить историю изменений

ЗАЧЕМ ИСПОЛЬЗОВАТЬ GIT

Работа в команде

Git позволяет разработчикам совместно работать над проектами, минимизируя конфликты

Отслеживание изменений

легко увидеть, кто и когда внес изменения, а также понять причины этих изменений

Восстановление предыдущих версий

возможность вернуться к рабочей версии проекта в случае ошибок или проблем

ПРЕИМУЩЕСТВА GIT

Безопасность

Все изменения сохраняются, что позволяет избежать потери данных

История изменений

Удобный доступ к истории проекта, что позволяет анализировать его развитие

Совместная работа

Упрощает процесс совместной работы, улучшая коммуникацию между членами команды

ОСНОВНЫЕ ПОНЯТИЯ GIT

Репозиторий `repository`

Хранилище, где хранится твой код вместе с историей всех изменений

Коммит `commit`

Сохраненная версия кода. Каждый раз, когда ты фиксируешь изменения, создаётся новый коммит с уникальным идентификатором

Ветка `branch`

Отдельная линия разработки. Например, можно создать ветку для тестирования новой функции, а потом объединить ее с основной веткой, когда всё будет готово

Слияние merge

Процесс объединения веток, когда изменения из одной ветки добавляются в другую

Клонирование clone

Копирование существующего репозитория на твой компьютер, чтобы начать работу с ним

Git позволяет каждому разработчику иметь полную копию проекта на своем компьютере. Это означает, что изменения могут быть внесены локально, а затем синхронизированы с удалённым репозиторием

Git использует три основных состояния для управления файлами:

01

Рабочая директория

место, где вы работаете над проектом

02

Индексация

staging area

промежуточное состояние, где вы добавляете изменения перед коммитом

03

Локальный репозиторий

хранит все ваши коммиты и историю изменений

ЛОКАЛЬНЫЙ И УДАЛЁННЫЙ РЕПОЗИТОРИИ

Локальный репозиторий

Хранится на вашем компьютере и содержит всю историю изменений

Удалённый репозиторий

Хранится на сервере (например, GitHub) и доступен для других разработчиков. Синхронизация между локальным и удалённым репозиториями осуществляется с помощью команд `push` и `pull`

ОСНОВНЫЕ КОМАНДЫ GIT

`git init`

Создает новый локальный репозиторий

`git add`

Добавляет изменения в индекс для следующего коммита

`git commit`

Сохраняет изменения в локальном репозитории с сообщением

ИТОГ

- Системы контроля версий, такие как Git, стали стандартом в разработке ПО благодаря своей надежности и удобству. Они обеспечивают безопасность, упрощают совместную работу и позволяют эффективно управлять проектами
- Принцип Git является основополагающим для эффективного управления проектами в разработке программного обеспечения
- Он упрощает отслеживание изменений, обеспечивает безопасность и стабильность работы команд