

Министерство образования Республики Беларусь
Учреждение образования

«Белорусский государственный университет
информатики и радиоэлектроники»

Факультет компьютерного проектирования
Кафедра инженерной психологии и эргономики

ПОЛЬЗОВАТЕЛЬСКИЕ ИНТЕРФЕЙСЫ ИНФОРМАЦИОННЫХ СИСТЕМ
ОТЧЁТ
по практической работе
«Основы Git»

Студент гр.210902

Рябцев Н.В.

Проверила

Давыдович К.И.

Минск 2024

Цель: изучение основ системы контроля версий Git с целью освоения её базовых функций для управления проектами, отслеживания изменений, совместной работы и организации рабочего процесса.

1. Инициализация репозитория

- `git init`

Создаёт новый локальный репозиторий в текущей папке.

2. Настройка пользователя

- `git config --global user.name "Ваше Имя"`
- `git config --global user.email "ваш_email@example.com"`

Настраивает имя и email, которые будут указываться в коммитах.

3. Добавление файлов в область подготовки (staging)

- `git add имя_файла`

Добавляет указанный файл в область подготовки.

- `git add .`

Добавляет все изменения в текущей директории в область подготовки.

4. Фиксация изменений (commit)

- `git commit -m "Описание изменений"`

Фиксирует изменения в репозитории с сообщением.

5. Просмотр статуса репозитория

- `git status`

Показывает текущий статус файлов: подготовлены они к коммиту, изменены или не отслеживаются.

6. Просмотр истории изменений

- `git log`

Выводит список всех предыдущих коммитов.

- `git log --oneline`

Показывает историю в кратком формате.

7. Работа с ветками

- `git branch`

Отображает список всех веток.

- `git branch имя_ветки`

Создаёт новую ветку.

- `git checkout имя_ветки`

Переключается на указанную ветку.

- `git checkout -b имя_ветки`

Создаёт и сразу переключается на новую ветку.

8. Слияние веток

- `git merge имя_ветки`

Объединяет указанную ветку с текущей.

9. Удаление ветки

- `git branch -d имя_ветки`
Удаляет указанную ветку.

10. Удалённые репозитории

- `git remote add origin ссылка_на_репозиторий`
Привязывает локальный репозиторий к удалённому (например, GitHub).
- `git push origin имя_ветки`
Отправляет изменения в удалённый репозиторий.
- `git pull origin имя_ветки`
Получает изменения из удалённого репозитория.

11. Клонирование репозитория

- `git clone ссылка_на_репозиторий`
Копирует существующий репозиторий из удалённого источника на локальный компьютер.

Вывод: Git – это мощный инструмент для контроля версий, который позволяет эффективно отслеживать изменения, организовывать командную работу и управлять проектами. Изучение базовых команд и принципов работы Git закладывает основу для более сложных сценариев использования, таких как работа с ветками, разрешение конфликтов и CI/CD.