Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

Кафедра инженерной психологии и эргономики

Дисциплина: Пользовательские интерфейсы информационных систем

Отчет

по практическому занятию

на тему:

«Git»

Проверил: Выполнил:

Студент группы 210902

Давыдович К.И. Кругликов П.А.

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** 
   1. *git init*

**Создаёт новый локальный репозиторий в текущей папке.**

1. **Настройка пользователя** 
   1. *git config --global user.name "Ваше Имя"*
   2. *git config --global user.email "ваш\_email@example.com"*

**Настраивает имя и email, которые будут указываться в коммитах.**

1. **Добавление файлов в область подготовки (staging)** 
   1. *git add имя\_файла*

**Добавляет указанный файл в область подготовки.**

* 1. *git add .*

**Добавляет все изменения в текущей директории в область подготовки.**

1. **Фиксация изменений (commit)** 
   1. *git commit -m "Описание изменений"*

**Фиксирует изменения в репозитории с сообщением.**

1. **Просмотр статуса репозитория** 
   1. *git status*

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

1. **Просмотр истории изменений** 
   1. *git log*

**Выводит список всех предыдущих коммитов.**

* 1. *git log --oneline*

**Показывает историю в кратком формате.**

1. **Работа с ветками** 
   1. *git branch*

**Отображает список всех веток.**

* 1. *git branch имя\_ветки* **Создаёт новую ветку.**
  2. *git checkout имя\_ветки*

**Переключается на указанную ветку.**

* 1. *git checkout -b имя\_ветки*

**Создаёт и сразу переключается на новую ветку.**

1. **Слияние веток** 
   1. *git merge имя\_ветки*

**Объединяет указанную ветку с текущей.**

1. **Удаление ветки** 
   1. *git branch -d имя\_ветки* **Удаляет указанную ветку.**
2. **Удалённые репозитории** 
   1. *git remote add origin ссылка\_на\_репозиторий*

**Привязывает локальный репозиторий к удалённому (например, GitHub).**

* 1. *git push origin имя\_ветки*

**Отправляет изменения в удалённый репозиторий.**

* 1. *git pull origin имя\_ветки*

**Получает изменения из удалённого репозитория.**

1. **Клонирование репозитория** 
   1. *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 и разрешение конфликтов.