

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»

Выполнил:

Студент  
гр.210902  
Трушко Р.С.

Проверил:

Давыдович К.И.

Минск 2024

# Git

Git — это распределенная система управления версиями, которая позволяет отслеживать изменения в файлах и координировать работу над проектами. Она популярна среди разработчиков и используется для управления кодом в большинстве современных программных проектов.

## Основные команды Git

### 1. **git init**

- Инициализация нового репозитория.
- Пример:

```
git init my_project
```

### 2. **git clone**

- Клонирование удаленного репозитория на локальную машину.
- Пример:

```
git clone https://github.com/user/repo.git
```

### 3. **git add**

- Добавление изменений в индекс (staging area).
- Пример:

```
git add filename.txt
```

- Или для добавления всех изменений:

```
git add .
```

### 4. **git commit**

- Сохранение изменений в локальном репозитории с сообщением.
- Пример:

```
git commit -m "Сообщение о коммите"
```

### 5. **git status**

- Проверка статуса рабочего каталога и индекса.
- Пример:

```
git status
```

### 6. **git push**

- Отправка локальных изменений в удаленный репозиторий.
- Пример:

```
git push origin main
```

## 7. **git pull**

- Получение и слияние изменений из удаленного репозитория.
- Пример:

```
git pull origin main
```

## 8. **git branch**

- Просмотр, создание или удаление веток.
- Пример для создания новой ветки:

```
git branch new-feature
```

## 9. **git checkout**

- Переключение между ветками или восстановление файлов.
- Пример:

```
git checkout new-feature
```

## 10. **git merge**

- Слияние одной ветки в другую.
- Пример:

```
git merge new-feature
```

## 11. **git log**

- Просмотр истории коммитов.
- Пример:

```
git log
```

## 12. **git reset**

- Отмена изменений и возврат к предыдущему состоянию.
- Пример:

```
git reset --hard HEAD~1
```

## **Заключение**

Git — мощный инструмент для управления версиями, который позволяет эффективно работать как индивидуально, так и в команде. Освоение основных команд Git — это важный шаг для любого разработчика.