

## Введение

Git Bash — это терминал для Windows, который предоставляет UNIX-подобную оболочку и инструменты командной строки. Его преимущество — поддержка стандартных git-команд в привычном виде.

---

### 1. Настройка пользователя

```
git config --global user.name "Student Example"
git config --global user.email "student@example.com"
```

#### Комментарий:

Эти команды задают имя и почту, которые будут записываться в каждый коммит. Ключ `--global` означает, что настройка сохранится для всех репозиториев на данном ПК.

---

### 2. Создание нового репозитория

```
mkdir demo-project
cd demo-project
git init
```

#### Комментарий:

- `mkdir` создаёт папку проекта.
  - `git init` превращает её в Git-репозиторий, добавляя скрытую папку `.git`.
- 

### 3. Проверка статуса

```
git status
```

### Комментарий:

Эта команда показывает, есть ли изменения в файлах, которые ещё не добавлены в систему контроля версий.

---

## 4. Добавление файлов и коммит

```
echo "Hello Git" > readme.txt  
git add readme.txt  
git commit -m "Добавлен файл readme.txt"
```

### Комментарий:

- `echo` создаёт файл.
  - `git add` подготавливает его к коммиту.
  - `git commit` фиксирует изменение в истории.
- 

## 5. Просмотр истории

```
git log --oneline
```

### Комментарий:

Отображает историю коммитов в сокращённом виде (короткий hash + сообщение).

---

## 6. Работа с ветками

```
git branch new-feature  
git checkout new-feature
```

### Комментарий:

- `git branch` создаёт ветку.
- `git checkout` переключает нас на неё. Теперь новые коммиты будут в этой ветке.

---

## 7. Слияние веток

```
git checkout main  
git merge new-feature
```

### Комментарий:

- Переключаемся на основную ветку.
- Выполняем слияние изменений из `new-feature`.

---

## 8. Подключение к удалённому репозиторию

```
git remote add origin https://github.com/student/demo-project.git  
git push -u origin main
```

### Комментарий:

- `git remote add origin` связывает локальный репозиторий с удалённым.
- `git push -u origin main` загружает коммиты в GitHub и устанавливает ветку `main` как отслеживаемую.

---

## Заключение

Git Bash позволяет выполнять все стандартные операции Git в привычной терминальной форме. Особенность этого клиента — удобство для пользователей Windows, поддержка unix-подобных команд ( `ls` , `cat` , `touch` ), что делает его универсальным инструментом для обучения и реальной разработки.

---