

# Информационные технологии (часть 2)

## 1. Требования к оформлению отчета

Отчет должен быть оформлен по СТП. Все рисунки должны быть читабельными. На каждом рисунке должна быть видна ФАМИЛИЯ СТУДЕНТА. **Обязательно выделяем команды на рисунках в терминале!**

## 2. Удаленные репозитории (часть 2)

### Введение

Целью данной лабораторной работы является изучение GitHub.

### Составить список используемых команд в терминале Linux

Команда	Значение	Расшифровать ключи

### Теория

Для создания копии репозитория на компьютере пользователя используется команда:

**git clone** репозиторий

Посмотреть список ссылок на удаленные репозитории можно с помощью команды:

**git remote**

Используя различные ключи, с помощью этой команды можно добавлять (ключи **add**), удалять (**rm**) или переименовывать ссылки на удаленные репозитории.

После клонирования в репозитории будет присутствовать одна локальная (**master**) и одна удаленная (**remote-tracking**) (**remotes/origin/master**) ветка. Посмотреть список удаленных веток можно с помощью команды **git branch** с ключом **-r**, для просмотра всех веток используется ключи **--all**.

Добавить удаленную ветку можно с помощью команды, аргументом которой является имя ветки в репозитории origin:

**git checkout** удаленная\_ветка

Удалить удаленную ветку (точнее, локальную копию удаленной ветки) можно с помощью обычной команды для работы с ветками (**git branch -d**).

В процессе работы на сервере могут появиться новые коммиты. Для синхронизации локальной копии с удаленной веткой используется команда:

**git fetch [удаленная\_ветка]**

С коммитами удаленной ветки нельзя работать напрямую, с помощью обычных команд **git**. Для того чтобы работать с загруженными в процессе синхронизации коммитами нужно провести слияние удаленной ветки с локальной. Сделать это можно с помощью обычной команды для слияния веток **git merge**.

Операции синхронизации ветки и ее слияния часто запускаются одна за другой, поэтому для запуска их последовательного выполнения в **git** имеется специальная команда:

**git pull**

Эта команда загружает с сервера и добавляет новые коммиты в ветку master. Использовать ее рекомендуется осторожно, поскольку возможны ошибки как на этапе загрузки коммитов в локальную копию удаленной ветки, так и на этапе слияния веток.

Обратная операция по загрузке коммитов из локальной ветки master в его серверную копию не разделяется на части, а выполняется одной командой:

**git push [репозиторий] [локальная\_ветка]**

При выполнении этой команды может возникать ошибка, связанная с тем, что с момента последней синхронизации в удаленной ветке на сервере добавлены коммиты. Исправить эту ошибку можно двумя способами.

Во-первых, можно выполнить синхронизацию и слияние веток, после чего повторно выполнить команду **git push**.

Во-вторых, можно выполнить команду **git push -f**. Ключ **-f** (от слова **force**) приведет к удалению с сервера коммитов, появившиеся после последней синхронизации, и выложит на сервер коммиты локальной ветки. Такой способ решения проблемы нужно применять с осторожностью, поскольку происходит удаление коммитов и может произойти потеря данных.

## **Задание к работе**

1) Создание репозитория на сервере **GitHub**:

1. зайти на сайт **GitHub.com** и залогиниться в свой аккаунт;
2. нажать на кнопку «**New**» в верхнем левом углу страницы;
3. в поле «**Repository name**» ввести транслитерацию своей фамилии;
4. оставить остальные настройки по умолчанию (репозиторий **Public**);
5. нажать на кнопку «**Create repository**».

2) Клонирование репозитория на компьютер два раза с именами **имя\_one** и **имя\_two**:

1. открыть терминал или командную строку на компьютере;

2. ввести команду «git clone

[https://github.com/<ваше\\_имя>/<название\\_репозитория>.git](https://github.com/<ваше_имя>/<название_репозитория>.git) имя\_one»;

3. ввести команду «git clone

[https://github.com/<ваше\\_имя>/<название\\_репозитория>.git](https://github.com/<ваше_имя>/<название_репозитория>.git) имя\_two»;

3) Синхронизация удаленной ветки и объединение веток:

1. открыть терминал или командную строку в папке с клонированным репозиторием **имя\_one**;

2. внести изменения в репозиторий (создать файл, изменить существующий и т.д.);

3. ввести команды «git add» и «git commit -m «Описание коммита»»;

4. ввести команду «git push» для отправки изменений на сервер;

5. перейти в терминал или командную строку в папке с клонированным репозиторием **имя\_two**;

6. ввести команду «git pull» для синхронизации удаленной ветки;

7. ввести команду «git merge origin/main» для объединения веток.

4) Добавление коммитов и разрешение конфликта:

1. открыть терминал или командную строку в папке с клонированным репозиторием **имя\_one**;

2. внести изменения в репозиторий (создать файл, изменить существующий и т.д.) и ввести команды «git add» и «git commit -m «Описание коммита»»;

3. ввести команду «git push» для отправки изменений на сервер;

4. перейти в терминал или командную строку в папке с клонированным репозиторием **имя\_two**

5. внести изменения в репозиторий (создать другой файл или изменить другой файл, чем в предыдущем коммите) и ввести команды «git add» и «git commit -m «Другое описание коммита»»;

6. Ввести команду «git pull» для синхронизации удаленной ветки;

7. Ввести команду «git merge origin/main» для объединения веток;

8. Разрешите возникший конфликт;

9. Выложите результат на сервер.