# Отчет по лабораторной работе №2.

Работу выполнил Шеожев Аслан Аскерович.

# Цель работы:

Изучить идеологию и применение средств контроля версий, а также освоить умения по работе с git.

# Задание:

Создать базовую конфигурацию для работы с git. Создать ключ SSH.

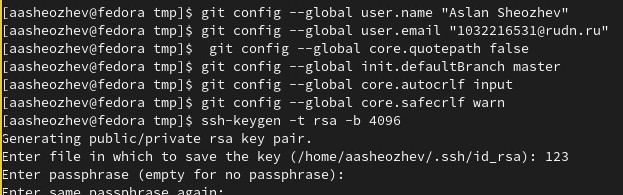
Создать ключ PGP. Настроить подписи git.

Зарегистрироваться на Github.

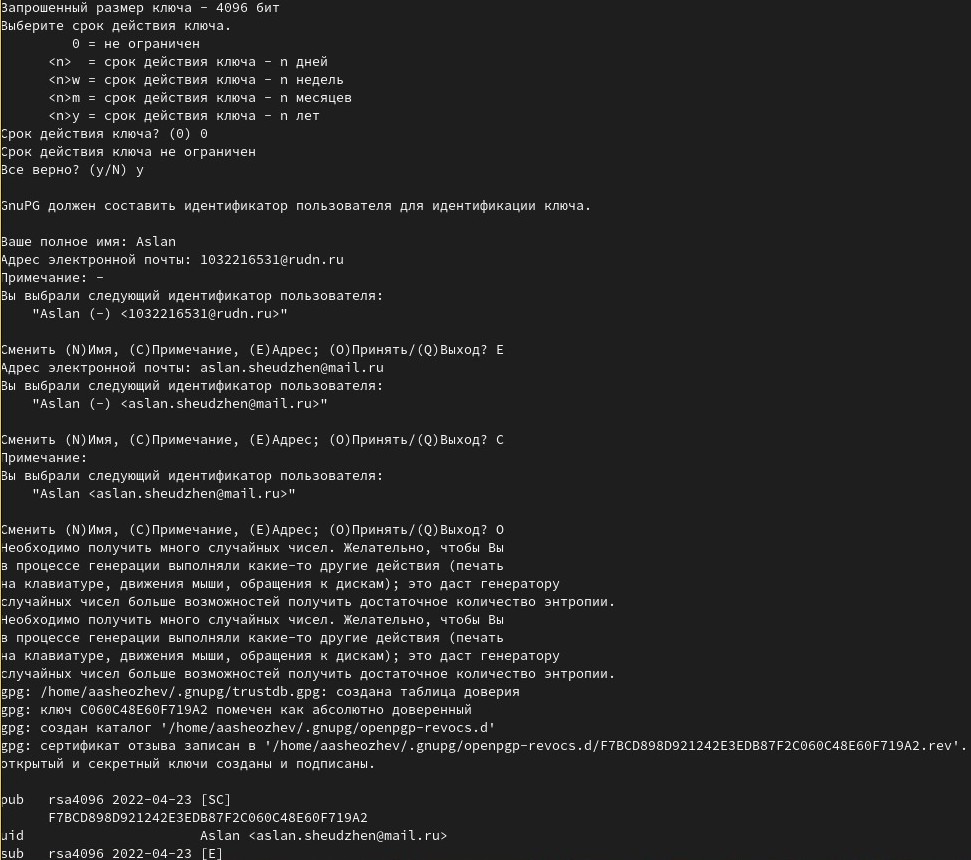
Создать локальный каталог для выполнения заданий по предмету.

# Ход работы:

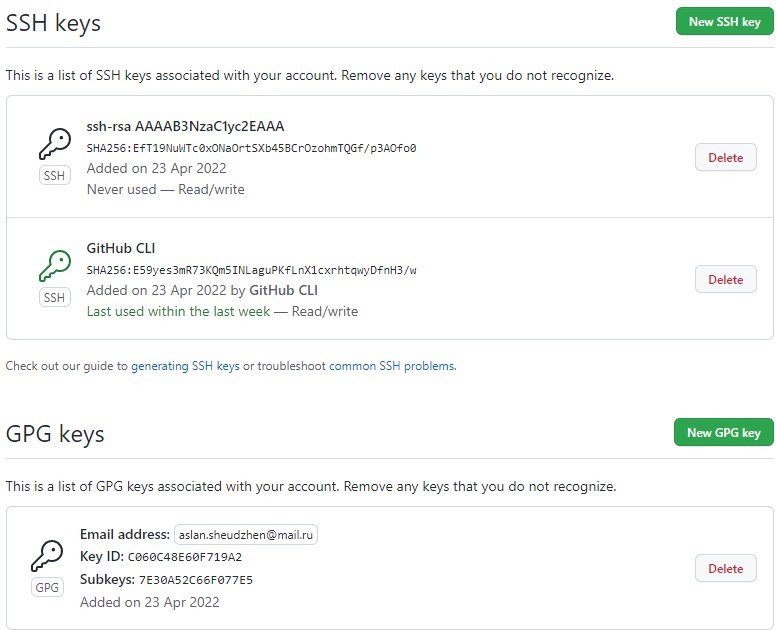
1. Ознакомился с основными командами git и установил все необходимое ПО (get-flow, gh). Настроил github. Создал учетную запись и заполнил основные данные.
2. Задал имя и email владельца репозитория, настроил utf-8 в выводе сообщений git, задал имя начальной ветки (будем называть её master). Создал ключи ssh по алгоритму rsa с ключём размера 4096 бит и по алгоритму ed25519.

{

width=100% } Рис. 1 (ввод основных сведений о программисте, а также настройка –utf8)

1. Создал ключи pgp. Тип – RSA and RSA. Размер – 4096. Срок действия не истекает никогда

{ width=100% } Рис. 2 (создал ключи gpg)

1. Добавил pgp и ssh ключи в github.

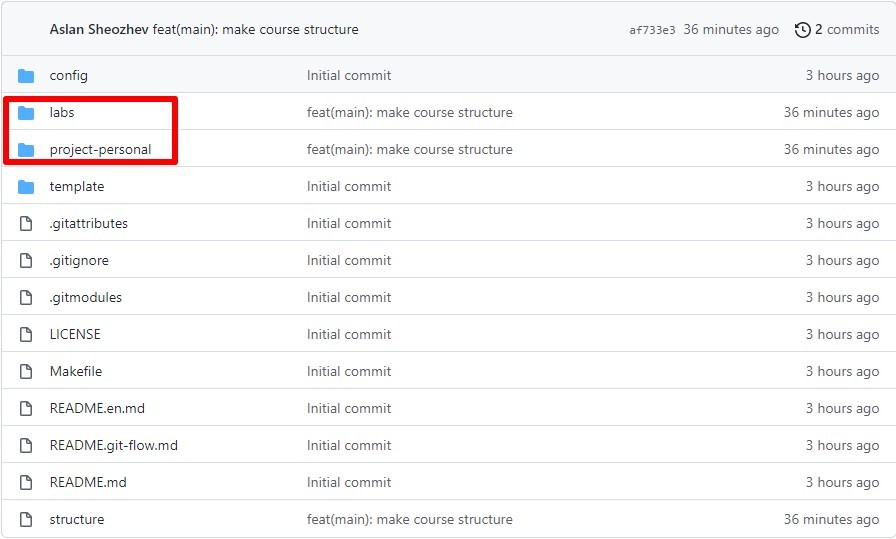
{ width=100% } Рис. 3 (введенные ключи на github)

1. Используя введенный email, указал Git применять его при подписи коммитов.

{ width=100% }

Рис. 4 (для подписи коммитов)

1. Приступил к настройке gh.
2. Создал шаблон рабочего пространства, по примеру из условия лабораторной работы. А после настроил каталог курса.



{ width=100% } Рис. 5 (результат на странице репозитория в github)

# Вывод:

Во второй лабораторной работе я изучил идеологию и применение средств контроля версий, освоил умения по работе с git.

# Контрольные вопросы:

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначаются?

Система контроля версий (VCS) нужны для работы нескольких людей над одним проектом, совместная работа путем изменения файлов в репозитории.

1. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия. хранилище – это пространство для хранения файлов;

commit – команда для записи индексированных изменений в репозитория;

история – в ней расположены все коммиты, по которым можно отследить автора, дату и сообщение;

рабочая копия – все файлы, кроме .git/ это рабочая копия.

1. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.

Централизованные системы контроля версий – проект и его файлы на 1 сервер, а децентрализованные системы контроля версия – при копировании данных данных удаленного репозитория, происходит полное копирование данных в локальный репозиторий.

1. Опишите действия с VCS при единоличной работе с хранилищем.

Создание репозиторий, добавление в них файлов, фиксация с помощью коммитов, изменение файлов репозитория.

1. Опишите порядок работы с общим хранилищем VCS.

Создание репозиториев, добавление файлов в него файлов, фиксация коммитами, изменение файлов репозитория, ожидание проверки коммитов.

1. Каковы основные задачи, решаемые инструментальным средством git?

Систематизация и возможность разработки параллельно.

1. Назовите и дайте краткую характеристику командам git.

Создание репозитория (git init), клонирование (gin clone), добавление изменений в индекс (git add), коммиты (gir commit), удаление файла (git rm), удаление изменений из индекса (git reset).

1. Приведите примеры использования при работе с локальным и удалённым репозиториями.

Для написания черновика по лабораторным работам я могу использовать репозитории.

1. Что такое и зачем могут быть нужны ветви (branches)?

Ветки служат для реализации параллельного программирования.

1. Как и зачем можно игнорировать некоторые файлы при commit?

Игнорирование нужно для неотслеживания файлов служебного типа.