Отчёт по лабораторной работе №2

Дисциплина: операционные системы

Логинов Георгий Евгеньевич

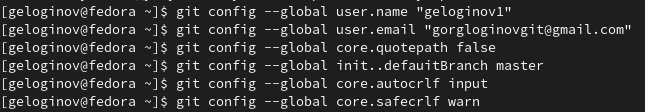
Содержание

# 1 Цель работы

Получить навыки по работе с удаленным репозиторием git.

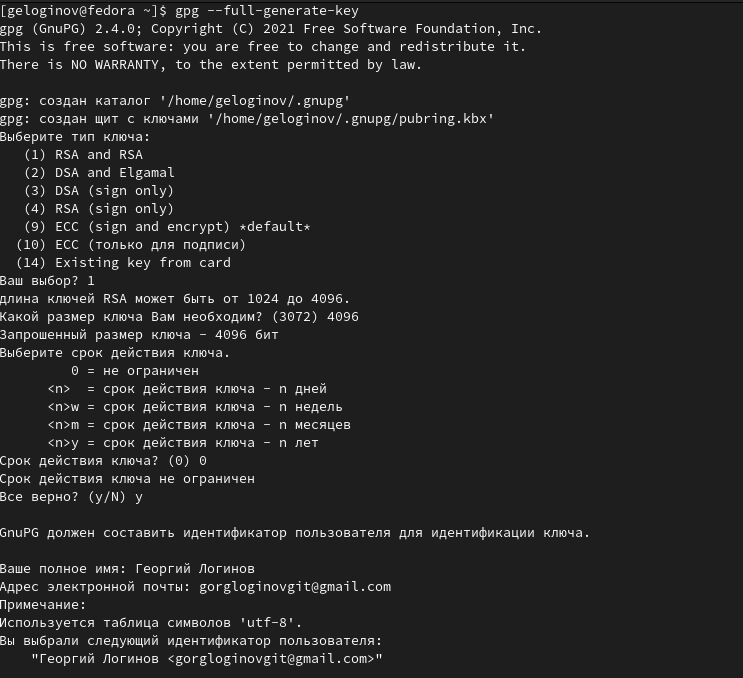
# 2 Выполнение лабораторной работы

Провожу базовою настройку git(рис. ??).



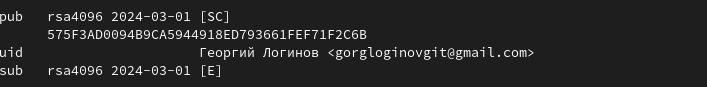
Базовая нстройка git

Генерирую ключ gpg(рис. ??).



Генерация ключа

Результат генерации(рис. ??).



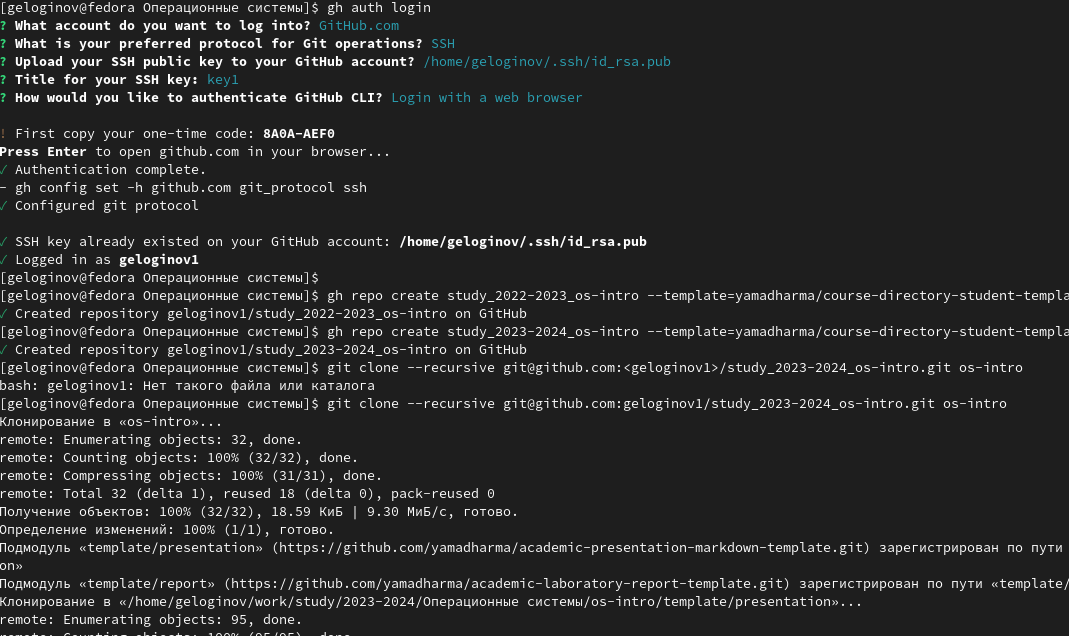
Результат генерации

Получаю ключ в ASCII формате для привязки к git(рис. ??).



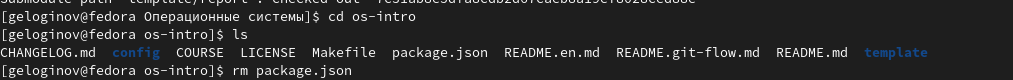
Ключ в ASCII

Создаю новый репозиторий в git(рис. ??).



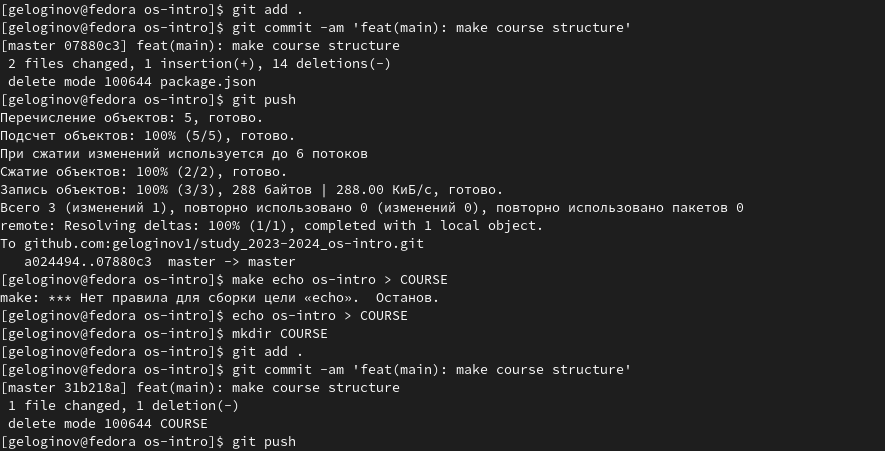
Создание репозитория

Удаляю лишний файл(рис. ??).



Удаление файла

Обновление данных в репозитории git(рис. ??).



Обновление данных

# 3 Контрольные вопросы

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначаются? Системы контроля версий позволяют хранить файлы не только на локальном компьютере, но и не локально. Это помогает в совместной работе с файлами или если нужно работать с нимим с нескольких устройств. Так же с их помощью можно создавать ветки (branches) которые позволяют сохранить рабочую версию и внести изменения только в копию.
2. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия. Хранилище содержит в себе всю информацию о проекте: историю, коммиты, все файлы Коммит – набор изменений и информация о них История – запись обо всех коммитах Рабочая копия – последняя версия, в которую вносятся изменения
3. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида. Централизованные VCS позволяют позльзователю подключиться ко всему хранилищу и запросить только одну конкретную версию Децентрализованные хранят в себе всю историю комммитов
4. Опишите действия с VCS при единоличной работе с хранилищем. Можно отправить измененную версию в репозиторий, склонировать существующий репозиторий, внести изменения, создать или удалить ветку
5. Опишите порядок работы с общим хранилищем VCS. Пользователь сконирует себе последнюю (или нужную ему) версию, редактирует ее, отправляет обратно в репозиторий
6. Каковы основные задачи, решаемые инструментальным средством git? Возможность работать командой, возможность вернуться к старым версиям, можно иметь несколько “путей развития” проекта (на разных ветках) и выбирать оптимальный по итогу
7. Назовите и дайте краткую характеристику командам git. git add . – сохранить изменения в текущем каталоге git commit -am 'anything' – создание коммита и с коментарием git push – отправка коммита
8. Приведите примеры использования при работе с локальным и удалённым репозиториями. В большом проекте разработчики будут клонировать локальный репозиторий себе на устройства, благодоря чему смогут работать параллельно.

* Если же репозиторий слишком большой, разработчики могут подключаться к нему удаленно и править код в новых ветках и т.п.

1. Что такое и зачем могут быть нужны ветви (branches)? Ветки (branches) которые позволяют сохранить рабочую версию и внести изменения только в копию. Это может быть нужно для избежания конфликта версий.
2. Как и зачем можно игнорировать некоторые файлы при commit? Обычно в игнорируемых файлах хранятся данные, которые не стоит выкладывать в общий доступ: пароли, ssh-ключи, базы данных.

# 4 Выводы

Были получены теоритические и практические навыки по работе с git и github.

# Список литературы