Лабораторная работа №2

Система контроля версий Git

Карпова Есения Алексеевна

Содержание

# 1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git

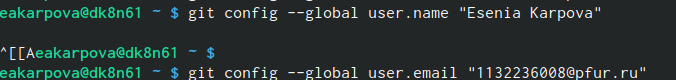
# 2 Задание

1. Настройка GitHub
2. Базовая настройка Git
3. Создание SSH-ключа
4. Создание рабочего пространства на основе шаблона
5. Создание репозитория курса на основе шаблона
6. Насторойка католога курса
7. Выполнение заданий для самостоятельной работы # Теоретическое введение

Системы контроля версий применяются при работе нескольких человек над одним проектом. При внесении изменений в содержание проекта система конитроля версий позволяет их фиксировать, совмещать изменения, произведенные разными участниками проекта, производить откат к любой более ранней версии проекта, если это потребуется

# 3 Выполнение лабораторной работы

1. Настройка GitHub Создаю учетную запись на сайте GitHub, заполнила основные данные учетной записи.
2. Базовая настройка Git Открываю консоль и делаю предварительную конфигурацию git. Ввожу команду git config –global user.name”“, уквазывая свое имя и команду git config –global user.email”“, указывая в ней свою электронную почту. (см. рис. ??)



Предварительеая конфигурация git

Настраиваю utf-8 в выводе сообщений git для корректного отображения символов(см. рис. ??)

Настраивание utf-8

Настраивание utf-8

Задаю имя «master» для начальной ветки(см. рис. ??)

Задаю имя “master”

Задаю имя “master”

Задаю параметр autocrlf со значением input, так как я работаю в системе Linux, чтобы конвертировать CRLF в LF только при коммитах (рис. 5). CR и LF – это символы, которые можно использовать для обозначения разрыва строки в текстовых файлах.(см. рис. ??)

Задаю параметр “autocrlf”

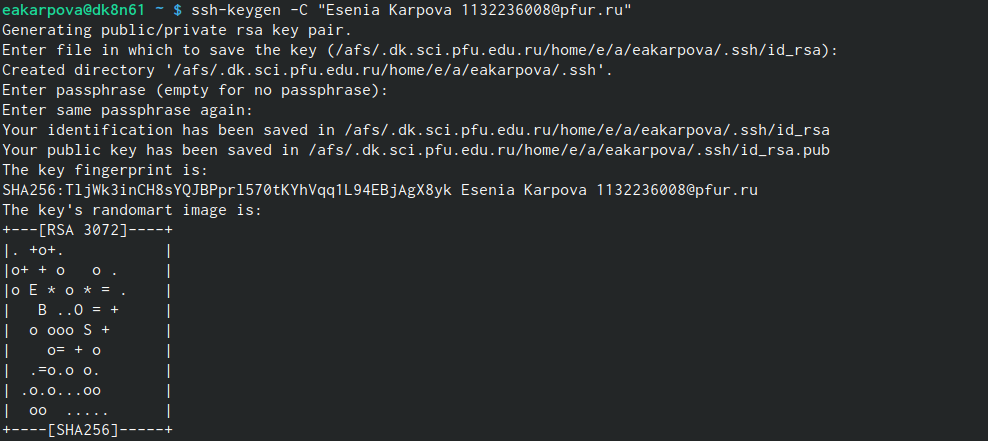
Задаю параметр “autocrlf”

Задаю параметр safecrlf со значением warn, так Git будет проверять преобразование на обратимость (рис. 6). При значении warn Git только выведет предупреждение, но будет принимать необратимые конвертации. (см. рис. ??)

Задаю параметр “safecrlf”

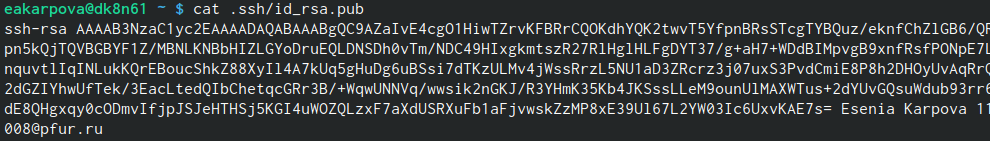
Задаю параметр “safecrlf”

1. Создание SSH-ключа Для последующей идентификации пользователя на сервере репозиториев необходимо сгенерировать пару ключей (приватный и открытый). Для этого ввожу команду ssh-keygen -C “Имя Фамилия, work@email”, указывая имя владельца и электронную почту владельца Ключ автоматически сохранится в каталоге ~/.ssh/(см. рис. ??)



Генерация SSH-ключа”

Далее необходимо загрузить сгенерированный открытый ключ. Для этого зайти на сайт под своей учётной записью и перейти в меню Setting . После этого выбрать в боковом меню SSH and GPG keys и нажать кнопку New SSH key . Скопировав из локальной консоли ключ в буфер обмена. Вставляем ключ в появившееся на сайте поле и указываем для ключа имя (Title).(см. рис. ??)



Копирование содержимого файла”

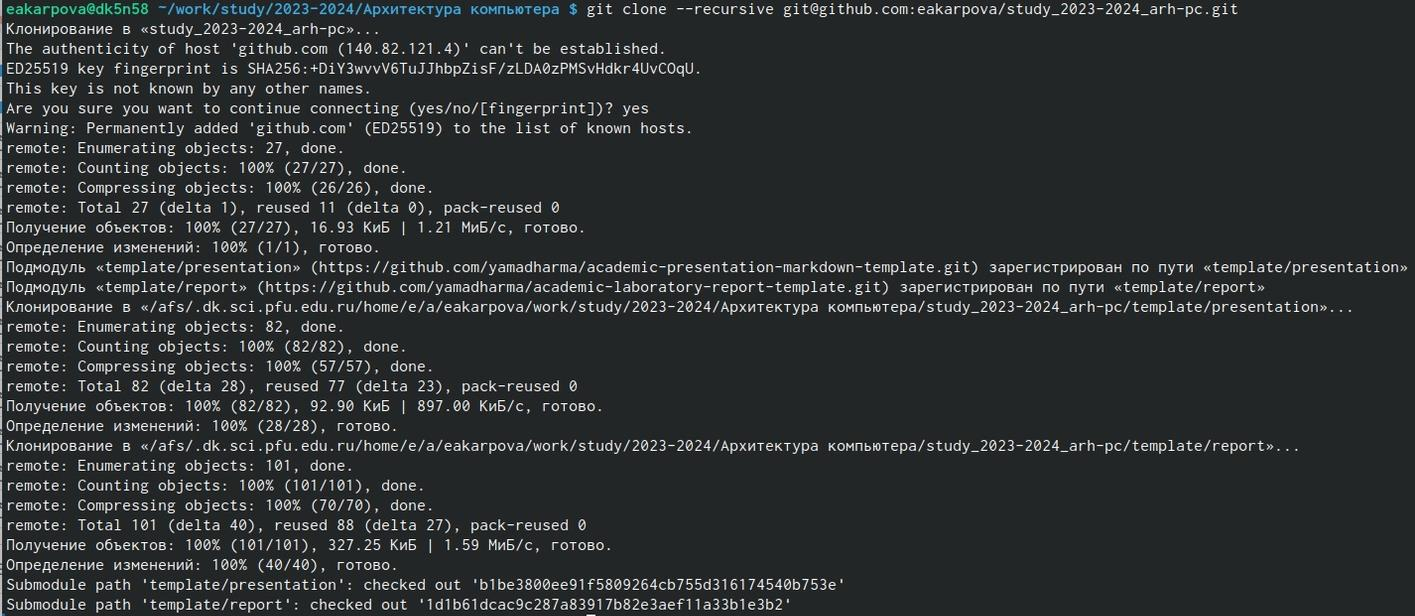
4.Создание рабочего пространства на основе шаблона Закрываю браузер, открываю терминал. Создаю директорию, рабочее пространство, с помощью утилиты mkdir, благодаря ключу - p создаю все директории после домашней ~/work/study/2022- 2023/“Архитектура компьютера” рекурсивно. Далее проверяю с помощью ls, действительно ли были созданы необходимые мне каталоги.

1. Создание репозитория курса на основе шаблона В браузере перехожу на страницу репозитория с шаблоном курса по адресу https://github.com/yamadharma/course-directory- student-template. Далее выбираю «Use this template», чтобы использовать этот шаблон для своего репозитория. В открывшемся окне задаю имя репозитория (Repository name): study\_2022–2023\_arh- pc и создаю репозиторий, нажимаю на кнопку «Create repository from template». Репозиторий создан.Через терминал перехожу в созданный каталог курса с помощью утилиты cd (см. рис. ??)

Перемещение между директориями”

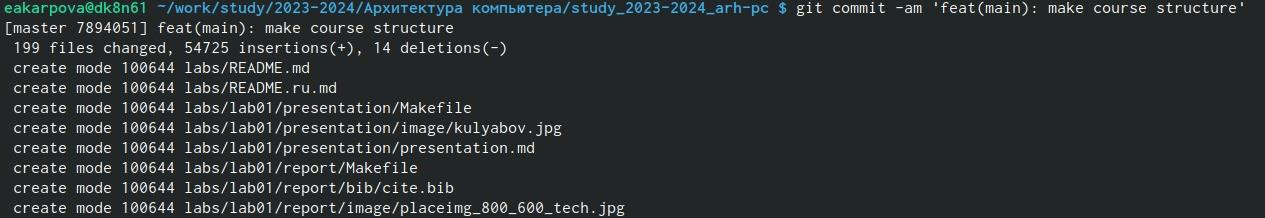
Перемещение между директориями”

Клонирую созданный репозиторий с помощью команды git clone –recursive git@github.com:/study\_2022–2023\_arh-pc.git arch- pc.(см. рис. ??)



Клонирование репозитория”

1. Настройка каталога курса Перехожу в каталог курса и удаляю лишние файлы с помощью rm. Создаю необходимые каталоги. Отправляю файлы на сервер с помощью команд (см. рис. ??)



Отправка файлов”

1. Задание для самостоятельной работы С помощью изученных в ходе первой лабораторной работы команд помещаю отчет по первой лабораторной работе в папку labs > lab02 > report Скопирую отчет по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства Перемещаю отчет по первой лабораторной работе в ~/lab01/report. Загружаю файл на github Ссылка: https://github.com/eakarpova/study\_2023-2024\_arh-pc/tree/master # Выводы

В ходе лабораторной работы я изучила идеологию и применение средств контроля версий, а также приобрела практические навыки по работе с системой git

# Список литературы

Лабораторная работа №2. Система контроля версий Git - Демидова А.В.