title: “Отчет по лабораторной работе №2” subtitle: “Архитектура компьютеров/Операционные системы” author: “Нилова Кристина Артуровна”

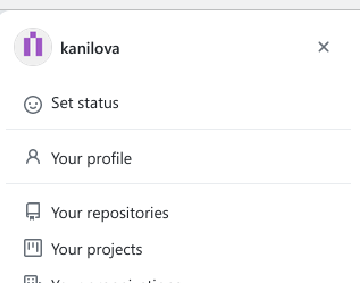
# Цель работы

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

# Ход работы

## 1. Настройка Github

1. Так как учетная запись на https://github.com/ ранее была создана, заходим уже в существующий аккаунт.



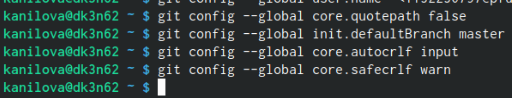
Аккаунт на Github

1. Запускаем виртуальную машину. Сначала сделаем предварительную конфигурацию git. При помощи команд git config –global user.name ““, git config –global user.email”“, указаем имя и email владельца репозитория.

Предворительная конфигурация git

Предворительная конфигурация git

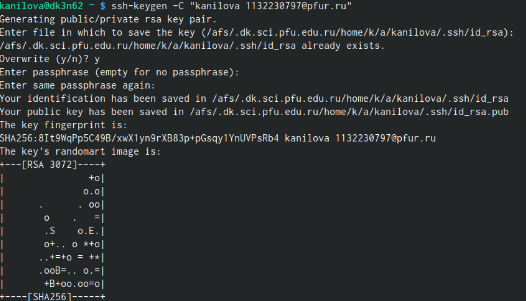
1. Настраиваем utf-8 в выводе сообщений git. Затем задаем имя начальной ветки (будем называть её master). Задаем параметр autocrlf со значением input (т.к. работаем в системе Linux, чтобы конвертировать CRLF в LF только при коммитах (рис. 1.3.)). CR и LF – это символы, которые используются для обозначения разрыва строки в текстовых файлах. Далее задаем параметр safecrlf со значением warn, так Git будет проверять преобразование на обратимость (рис. 1.3)



Настройка кодировки, создание имени для начальнойветки и выполнение параметров: autocrlf, safecrlf

## 2. Создание SSH ключа.

1. Для последующей идентификации пользователя на сервере репозиториев необходимо сгенерировать пару ключей (приватный и открытый). Вводим команду ssh-keygen -C “Имя Фамилия, work@email”, указывая имя и корпоративную почту владельца. Ключи сохраняются в каталоге ~/.ssh/.(рис.1.4).



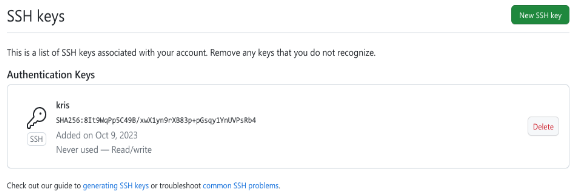
Создание SSH ключа

1. Используем xclip утилиту, позволяющую скопировать любой текст через терминал. Для начала, в дистрибутиве Linux Ubuntu, ее необходимо установить с помощью команды apt-get install с ключом -y от имени суперпользователя, введя в начале команды sudo (рис. 1.5.). Копируем открытый ключ из директории, в которой он был сохранен, с помощью утилиты xclip

Утилита xclip.

Утилита xclip.

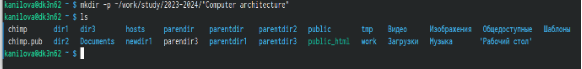
1. Открываем браузер, заходим на сайт https://github.com/ . Открываем профиль и выбираем страницу «SSH and GPG keys» кнопку «New SSH key» . Вставляем скопированный ключ в поле «Key». В поле Title указываем имя для ключа. Нажимаем «Add SSH-key», чтобы завершить добавление ключа (рис. 1.6.).



Добавление ключа

## 3. Сознание рабочего пространства и репозитория курса на основе шаблона

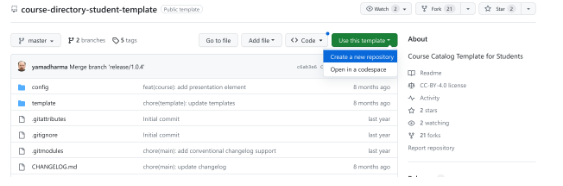
1. Открываем терминал. Создаем директорию, рабочее пространство, с помощью утилиты mkdir, благодаря ключу -p создаем все директории после домашней ~/work/study/2023-2024/“Computer architecture” рекурсивно. Далее проверяем с помощью ls, создание необходимых каталогов (рис. 1.7.).



Создание рабочего пространства

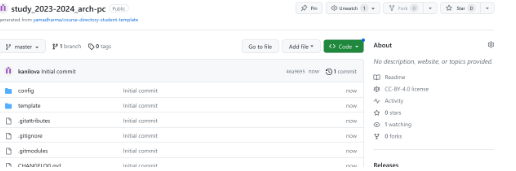
## 4. Создание репозитория курса на основе шаблона

1. В браузере переходим на страницу репозитория с шаблоном курса по адресу https://github.com/yamadharma/course- directory-student-template. Далее выбираем «Use this template», чтобы использовать этот шаблон для репозитория (рис. 1.8.).



Страница репозитория с шаблоном курса.

1. Задаем имя репозитория (Repository name): study\_2022– 2023\_arch- pc и создаем репозиторий, нажав на кнопку «Create repository» (рис. 1.9.)



Созданный репозиторий.

1. Через терминал переходим в созданный каталог курса с помощью утилиты cd (рис. 1.10.).

Перемещение между директориями.

Перемещение между директориями.

1. Копируем ссылку для клонирования на странице созданного репозитория, сначала перейдя в окно «code», далее выбрав в окне вкладку «SSH» (рис. 1.11.)

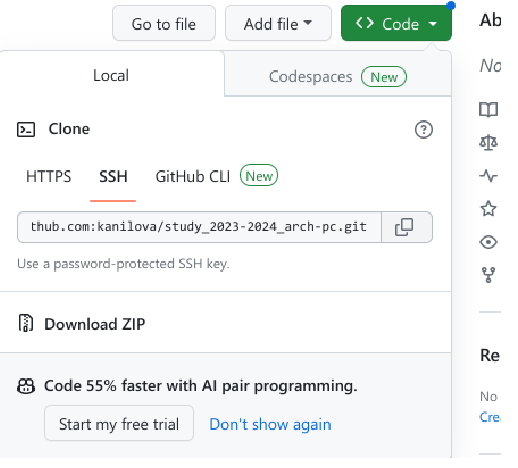


Рис. 1,11

1. Клонируем созданный репозиторий с помощью команды git clone –recursive git@github.com:/study\_2023–2024\_arh-pc.git arch-pc (рис. 1.12.)

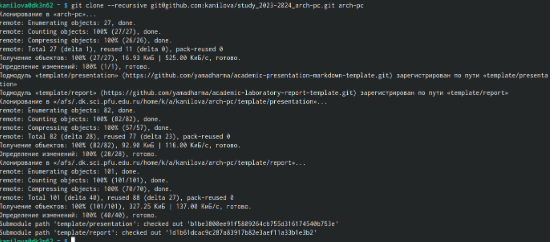


Рис. 1,12

## 5. Настройка каталога курса

1. Переходим в каталог arch-pc с помощью утилиты cd (рис. 1.13.).

Рис. 1,13

Рис. 1,13

1. Удаляем лишние файлы с помощью утилиты rm. Создаем необходимые каталоги (рис. 1.14).

Рис. 1,14

Рис. 1,14

1. Отправляем созданные каталоги с локального репозитория на сервер: добавляем все созданные каталоги с помощью git add, комментируем и сохраняем изменения на сервере, как добавление курса с помощью git commit (рис. 1.15.).

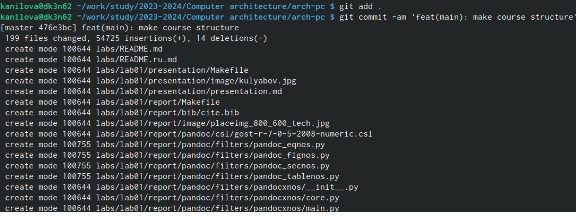


Рис. 1,15

1. Отправляем все на сервер с помощью push (рис. 1.16.).

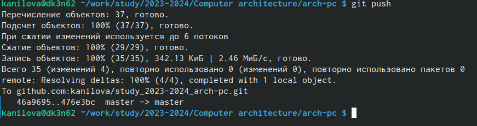


Рис. 1,16

1. Проверяем правильность выполнения работы на самом сайте GitHub (рис.1.17.).

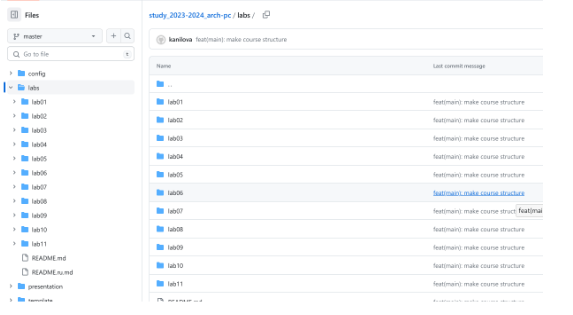


Рис. 1,17

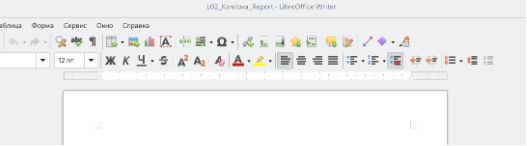
## 6. Самостоятельная работа

1. Переходим в директорию labs/lab02/report с помощью утилиты cd. Создаем в каталоге файл для отчета по второй лабораторной работе с помощью утилиты touch (рис.1.18.).

Создание файла

Создание файла

1. Оформляем отчет в текстовом процессоре LibreOffice Writer. После открытия текстового редактора, открываем в нем созданный файл.
2. Переходим из подкаталога lab02/report в подкаталог lab01/report с помощью утилиты cd (рис. 1.19.).



Перемещение между директориями

1. Проверяем местонахождение файлов с отчетами по первой лабораторной работe. Она должна быть в подкаталоге домашней директории «Documents», для проверки используtv команду ls (рис. 1.20.).

Проверка местонахождения файлов

Проверка местонахождения файлов

1. Добавлем файл L01\_Kanilova\_Report.pdf. Сохраняем изменения на сервере командой git commit -m “…”. То же самое делаем для отчета по второй лабораторной работе. Переходим в директорию labs/lab02/report с помощью cd, добавляем с помощью git add нужный файл, сохраняем изменения с помощью git commit (рис. 1.22./1.23.)

Рис. 1.22

Рис. 1.22

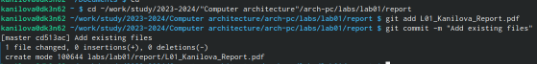


Рис. 1.23

1. Отправляем в центральный репозиторий сохраненные изменения командой git push -f origin master (рис. 1.24.)



Отправка изменений в центральный репозиторий

1. Проверяем на сайте https://github.com/ правильность выполнения заданий.(рис. 1.25.)

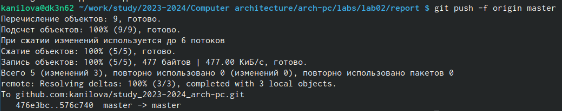


Рис. 1.25

1. Просматриваем изменения. Видим, что добавлены файлы с отчетами по лабораторным работам.(рис. 1.26.).

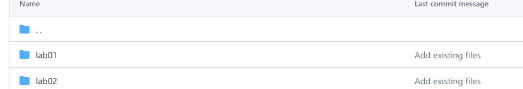
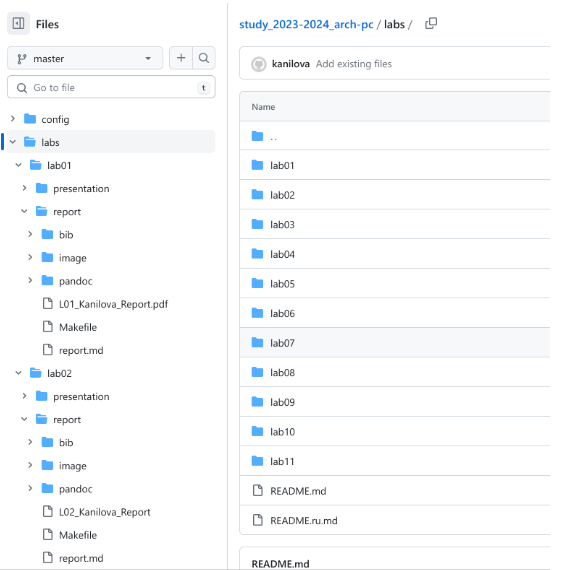


Рис. 1.26

1. Проверяем, находятся ли отчеты по лабораторным работам в нужных каталогах репозитория. (рис.1.27.)



Каталог lab01/report.

# Вывод

В ходе выполнения лабораторной работы изучены идеология и применение средств контроля версий. Приобретены практические навыки по работе с системой git.