Лабораторная работа №2

Первоначальна настройка git

Чесноков Артемий Павлович НПИбд-02-22

Содержание

# 1 Цель работы

Изучить идеологию и применение средств контроля версий и Освоить умения по работе с git.

# 2 Выполнение лабораторной работы

1. С помощью команды sudo apt git установим git + config настройка (рис. [[1](#fig:001)]).

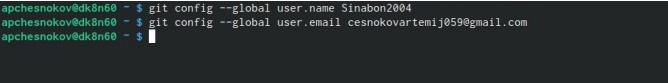


Figure 1: Зайшли на сайт

1. Зайдем на официальный сайт Github (рис. [[2](#fig:002)]).

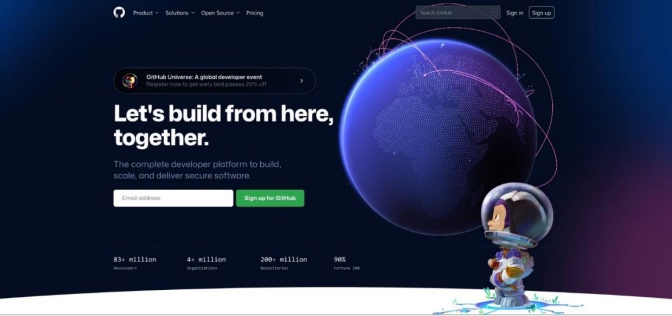


Figure 2: Зайшли на сайт

1. Проходим все этапы регистрации (рис. [??]).



Figure 3: Зарегистрировались

1. Настраиваем систему (рис. [[4](#fig:004)]).

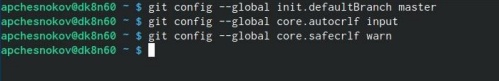


Figure 4: Работа с командной строкой

1. Продолжаем настривать систему (рис. [[5](#fig:005)]).

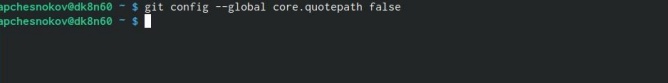


Figure 5: Выполняем указанные команды

1. Генерируем SSH ключ (рис. [[6](#fig:006)]).

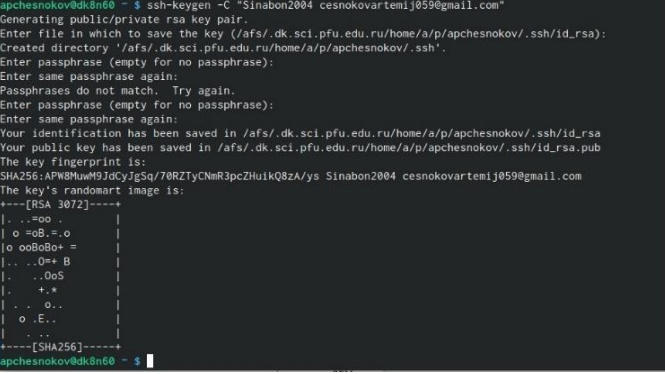


Figure 6: Выполняем указанные команды

1. Копируем ключ (рис. [[7](#fig:007)]).

Figure 7: используем эту команду, чтобы скопировать

Figure 7: используем эту команду, чтобы скопировать

1. Вставляем ключ в Github (рис. [[8](#fig:008)]).

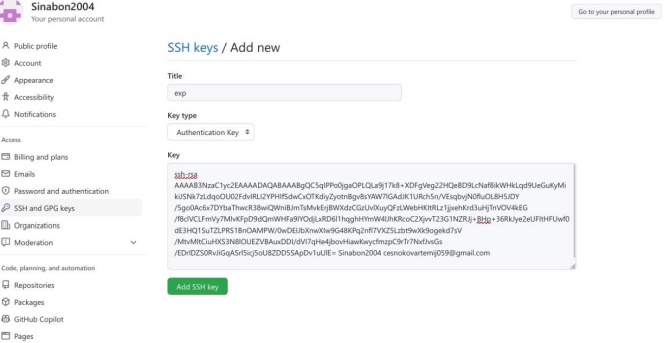


Figure 8: так это выглядит

1. Создаем PGP ключ (рис. [[9](#fig:009)]).

Figure 9: этими командами

1. Копируем ключ для Github (рис. [[10](#fig:010)]).

Figure 10: этими командами

1. Добавляем ключ на Github аналогично SSH ключу в настройках.
2. Авторизируемся в gh по командной строке (рис. [[11](#fig:011)]).

Figure 11: этими командами

1. Создаем репозиторий курса на основе шаблона (рис. [[12](#fig:012)]).

Figure 12: этими командами

1. Настраиваем каталог курса (рис. [[13](#fig:013)]).

Figure 13: этими командами

1. git push (рис. [[14](#fig:014)]).

Figure 14: этими командами

#Контрольные вопросы 1. Системы контроля версий = место для хранения кода. Предназначена для хранения кода

1. Хранилище = место хранения всех версий. commit = создание новой версии. история = история изменения кода. рабочая копия = текущее состояние файлов проекта.
2. У централизованныых одно хранилище для всего проекта
3. Пользователь размещает новую версию и может вернуться к старым версиям.
4. Общее хранилище работает на много пользователей, поэтому изменять версии и общий код нужно иначе.
5. git - хранит информацию об изменениях, командная работа над кодом.
6. git add = добавляет содержимое рабочей директории а индекс. git commit = сохраняет слепок в базе данных. git rm = удаление файлов из индекса. git mv = перемещиение файлов git clean =удаление мусора.
7. Если хотим поделиться наработками, то отправляем код на удаленный репоизиторий.
8. Ветки нужны, чтобы несколько программистов могли вести работу над одним и тем же проектом, не мешая друг другу.
9. Это упрощает работу, ибо не все файлы обязательно коммитить.

# 3 Выводы

Научились пользоваться git. # Список литературы{.unnumbered}