Некоммерческое акционерное общество

Карагандинский технический университет имени Абылкаса Сагинова

Кафедра ИВС

**Лабораторная работа №4**

**Дисциплина: «**Современные технологии программирования»

**Тема:** «**Управление Версиями Проекта**»

**Подготовили:**

Студенты группы

ИСМ-23-2

Авдиев М.Т. и

Дашкин И.Р.

**Приняла:**

Томилова Н.И.

Караганда 2023

**Лабораторная работа № 4**

*Тема:* Управление Версиями Проекта

*Цель работы:*приобрести практические навыки работы с системой контроля версий Git

***Лабораторная работа*** состоит в изучении теоретического материала и получении практических навыков при создании серверов  в Python. Сдача лабораторной работы заключается в ответах на контрольные вопросы и работой с серверами в python.

Вариант 1:

- Создать

1 Фоpматиpование документов.

Пpогpамма печати (вывода в дpугой файл) текстового файла.

Режимы:

 - ввод имени файла,

 - вывод на принтер или в дpугой файл или на экpан,

 - задание числа стpок на стpанице печати,

 - задание числа пpобелов слева на стpанице,

 - печать текущей даты и вpемени,

 - печать имени файла, даты, вpемени .

Задачи:

1.     Создание локальной копии главного репозитория:

Git clone https://Github.com/pupkin/test

cd test *(не забыть перейти в каталог репозитория)*

2.     Добавление новых файлов в репозиторий. Избранные файлы:

Git add file1 file2 file3

Все новые файлы:

Git add .

3.     Сохранение изменений файлов:

it commit ­am "commit description"

4.     Получение изменений из главного репозитория:

​Git pull

5.     Отправка изменений в главный репозиторий (с авторазрешением конфликтов):

Git pull *(проверить на наличие новых изменений в репозитории и,* *если* *они есть, выкачать их и объединить с локальными изменениями)*

Git push *(отправить изменения в репозиторий)*

Вариант задания 3:

Работа со стpоками и текстовыми файлами. Статистический анализ текста.

Ход работы:

1. Инициализация проекта командой $ git init (рисунок 1)

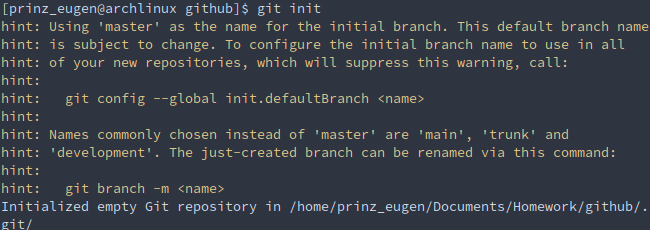


Рисунок 1 – инициализация.

2. Добавление новых файлов в репозиторий командой git add

$ git add first.py mbox.txt FileAnalyzer.py

3. Добавление комментария командой git commit:

$ git commit -m "first commit"

4. Выбор основного бранча командой git branch:

$ git branch -M main

5. Отправка файлов на гитхаб командой git push:

$ git push -u origin main

6. Удаление из репозитория происходит на сайте гитхаба (рисунок 2)

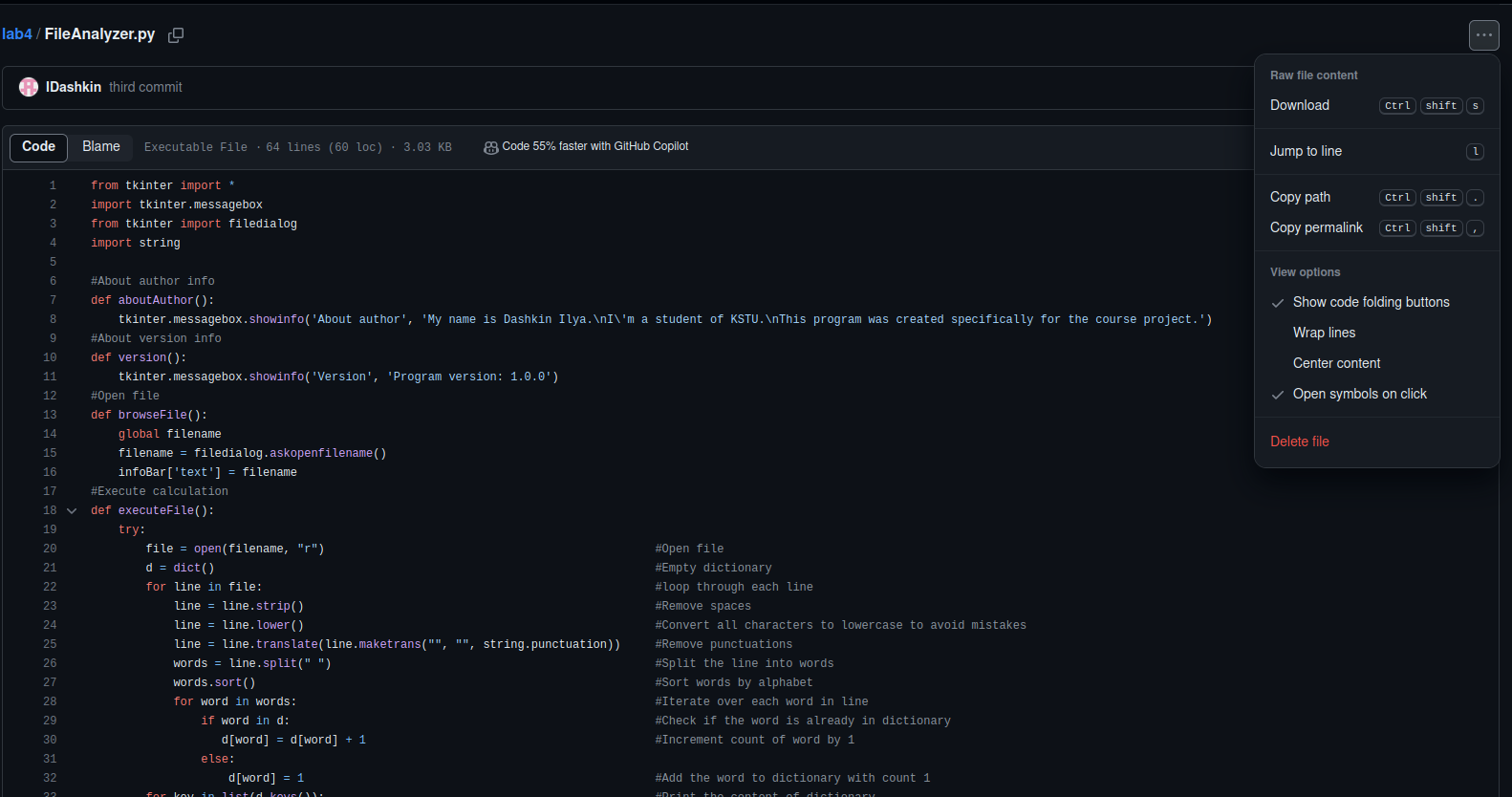


Рисунок 2 – удаление файла из репозитория.

7. Получение изменений из репозитория командой git pull (рисунок 3):

$ git pull origin main

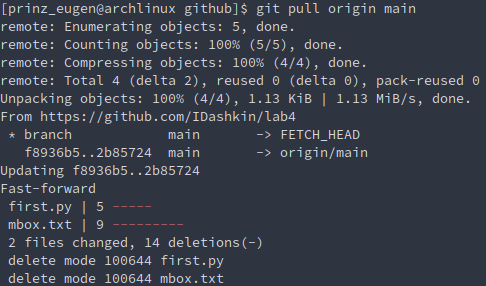


Рисунок 3 – синхронизация файлов.

**Контрольные вопросы**

1. Что такое система контроля версий

программное обеспечение для облегчения работы с изменяющейся информацией. Система управления версиями позволяет хранить несколько версий одного и того же документа, при необходимости возвращаться к более ранним версиям, определять, кто и когда сделал то или иное изменение, и многое другое. Такие системы наиболее широко используются при разработке программного обеспечения для хранения исходных кодов разрабатываемой программы. Однако они могут с успехом применяться и в других областях, в которых ведётся работа с большим количеством непрерывно изменяющихся электронных документов.

1. Зачем нужна система контроля версий

Программное обеспечение контроля версий отслеживает все вносимые в код изменения в специальной базе данных. При обнаружении ошибки разработчики могут вернуться назад и выполнить сравнение с более ранними версиями кода для исправления ошибок, сводя к минимуму проблемы для всех участников команды.

1. Значение команды Add

Команда git add добавляет содержимое рабочего каталога в индекс (staging area) для последующего коммита. По умолчанию git commit использует лишь этот индекс, так что вы можете использовать git add для сборки слепка вашего следующего коммита.

Это одна из ключевых команд Git, мы упоминали о ней десятки раз на страницах книги. Ниже перечислены наиболее интересные варианты использования этой команды.

1. Значение команды Commit

Команда git commit берёт все данные, добавленные в индекс с помощью git add, и сохраняет их слепок во внутренней базе данных, а затем сдвигает указатель текущей ветки на этот слепок.

Вы познакомились с основами модели коммитов в разделе [Коммит изменений](https://git-scm.com/book/ru/v2/ch00/r_committing_changes) главы 2. Там же мы продемонстрировали использование опций -a для добавления всех изменений в индекс без использования git add, что может быть удобным в повседневном использовании, и -m для передачи сообщения коммита без запуска полноценного редактора.

1. Значение команды Push

Команда git push используется для установления связи с удалённым репозиторием, вычисления локальных изменений отсутствующих в нём, и собственно их передачи в вышеупомянутый репозиторий. Этой команде нужно право на запись в репозиторий, поэтому она использует аутентификацию. Несколько примеров использования git push для совместной работы в нескольких удалённых репозиториях одновременно.

1. Значение команды Pull

Команда git pull работает как комбинация команд git fetch и git merge, т. е. Git вначале забирает изменения из указанного удалённого репозитория, а затем пытается слить их с текущей веткой.

1. Как создать новый репозиторий

Для создания нового репозитория используется команда git init . Команду git init выполняют только один раз для первоначальной настройки нового репозитория. Выполнение команды приведет к созданию нового подкаталога . git в вашем рабочем каталоге.

Обычно вы получаете репозиторий Git одним из двух способов:

1. Вы можете взять локальный каталог, который в настоящее время не находится под версионным контролем, и превратить его в репозиторий Git, либо
2. Вы можете **клонировать** существующий репозиторий Git из любого места.

В обоих случаях вы получите готовый к работе Git репозиторий на вашем компьютере.

для Windows:

$ cd C:/Users/user/my\_project

а затем выполните команду:

$ git init

1. Что такое локальный репозиторий

Локальный — расположен на одном компьютере, и работать с ним может только один человек. локальный репозиторий - репозиторий на пк, где происходит разработка и фиксируются изменения, которые отправляются на удалённый репозиторий, то есть по сути это и есть та же самая папка с файлами проекта на нашем п

9.                                  Что такое удаленный репозиторий

Удалённый репозиторий — репозиторий, находящийся на удалённом сервере. Это общий репозиторий, в который приходят все изменения и из которого забираются все обновления.

Удалённые репозитории представляют собой версии вашего проекта, сохранённые в интернете или ещё где-то в сети. У вас может быть несколько удалённых репозиториев, каждый из которых может быть доступен для чтения или для чтения-записи.