# 1. ЗАДАНИЕ

Используя сервис управления версиями Git, коллективно создайте и отредактируйте два фрагмента кода, каждый объемом не более 20 строк. При работе над каждым фрагментом кода должен быть свой вариант распределения ролей «ведущий разработчик» и «разработчик». При выполнении работы используйте данные методические указания к учебной практике «Управление версиями на примере сервиса Git». Отчет о выполненном задании выложите в портал.

## 1.1. Цель работы

* Изучить возможности системы управления версиями и коллективной разработкой программного проекта.
* Освоить методы работы с утилитой Git для управления версиями и коллективной разработкой программного проекта.

## 1.2. Средства выполнения

* Приложение Git.

## 1.3. Пункты задания для выполнения

* Запустить консоль git. Создать новый репозиторий (в папке по фамилии студента).
* Добавить в папку репозитория файлы (3 python файла, каждый из которых представляет решение задачи из 7 варианта). Зафиксировать состояние репозитория (выполнить commit).
* Внести изменения в файлы. Зафиксировать новое состояние репозитория.
* Создать новую ветку 1. Внести в нее изменения (добавить новый файл и изменить существующий файл: добавить, удалить и изменить строки) и зафиксировать их.
* Переключиться на ветку мастера. Внести в нее изменения (добавить новый файл; изменить существующие файлы: добавить, удалить и изменить строки первоначального файла) и зафиксировать их.
* Продемонстрировать слияние веток. Разрешить возникший конфликт. Просмотреть дерево изменений веток (историю).
* Продемонстрировать откат изменений в ветке 1.
* Создать удаленный репозиторий (на github.com).
* Отправить данные на удаленный репозиторий (выполняется одним из студентов подгруппы – ведущий разработчик). Добавить к удаленному репозиторию второго участника проекта (разработчик).
* Получить данные из удаленного репозитория (выполняется каждым из участников проекта).
* Изменить полученные данные.
* Зафиксировать изменения и отправить их на удаленный репозиторий (выполняется каждым из участников проекта).
* Получить данные из удаленного репозитория.
* Просмотреть историю изменений.
* Повторить все описанные шаги с другим участником команды в качестве ведущего разработчика.

## 1.4. Дополнительное задание

* Продемонстрировать работу revert и reset.
* Продемонстрировать работу rebase.