1. Выберите тематику программы, которую собираетесь написать. Создайте для нее рабочую директорию

Изображение выглядит как текст, снимок экрана, монитор

Автоматически созданное описание

1. Инициализируйте в рабочей директории репозиторий при помощи команды git init.

Изображение выглядит как текст

Автоматически созданное описание

1. Выполните в репозитории команду git status. Проинтерпретируйте полученное сообщение.

Изображение выглядит как текст

Автоматически созданное описание

Пока что у нас нет никаких коммитов, мы на ветке master

1. Создайте файл для исходного текста программы. Выполните команду git status.

Изображение выглядит как текст

Автоматически созданное описание

1. Добавьте созданный файл под версионный контроль при помощи команды git add. Еще раз выполните git status.

Изображение выглядит как текст

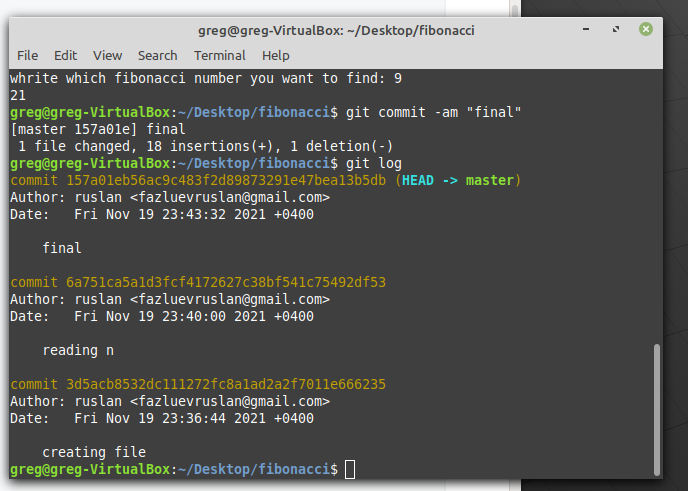
Автоматически созданное описание

1. Сделайте начальный коммит при помощи команды git commit с опцией -m.

Изображение выглядит как текст

Автоматически созданное описание

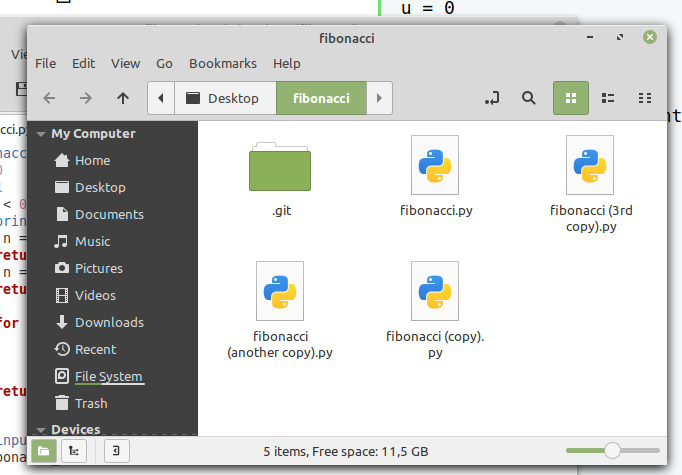
1. Сделайте еще несколько коммитов. Выполните команду git log для просмотра истории коммитов.



1. Сделайте так, чтобы при коммите измененные файлы автоматически добавлялись в коммит.

Нужно использовать опцию -am

1. Добавьте еще несколько файлов с исходным текстом программы.



1. Добавьте все новые файлы под версионный контроль одной командой.

Изображение выглядит как текст

Автоматически созданное описание

1. На всех стадиях работы пользуйтесь командой git status.

Изображение выглядит как текст

Автоматически созданное описание

1. Инициализируйте в рабочей директории виртуальное окружение (Если вы пишите не на Python, то можете инициализировать какой-либо программный фреймворк, либо начать работать в IDE, которая создает скрытую папку с настройками в рабочем каталоге).

Изображение выглядит как текст

Автоматически созданное описание

1. Добавьте созданную служебную папку в файл .gitignore. Проверьте, что они не добавляются в репозитории при добавлении новых файлов с исходным кодом.

Изображение выглядит как текст

Автоматически созданное описание

1. Создайте новую тематическую ветку git branch. Перейдите в нее с помощью git checkout. Выведите на экран список всех веток.

Изображение выглядит как текст

Автоматически созданное описание

1. Сделайте несколько коммитов в основную и тематическую ветки.

Изображение выглядит как текст

Автоматически созданное описание

1. Слейте изменения в основную ветку с помощью git merge. Если произошел конфликт слияния, разрешите его и завершите слияние с помощью git commit.

Изображение выглядит как текст

Автоматически созданное описание

1. При получении в процессе разработки программы в стабильно работающем состоянии, слейте это состояние в основную ветку и добавьте к коммиту слияния пометку с номером релиза.

Изображение выглядит как текст

Автоматически созданное описание