# Работа с Git

**Первичная настройка** При первом использовании Git требуется задать идентификационные данные пользователя. Выполняются две команды терминала (Рис. 1) для регистрации имени и электронного адреса.

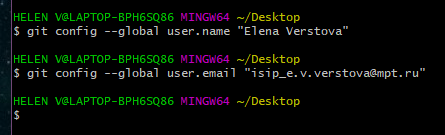


Рисунок 1 - Установка имени и электронной почты

Далее настраивается обработка символов перевода строки (Рис. 2)

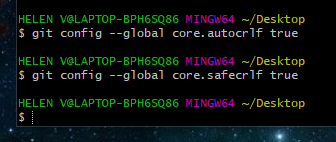


Рисунок 2 - Параметры окончаний строк

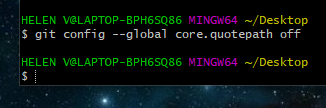
и параметры отображения Unicode-символов (Рис. 3).

Рисунок 3 - Установка отображения unicode

1. **Инициализация проекта** Создайте пустую директорию и поместите в нее файл hello.html (Рис. 4)

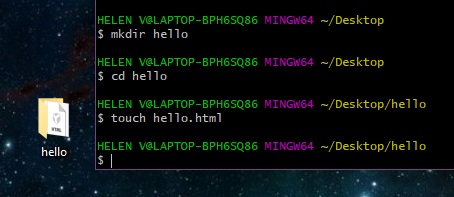


Рисунок 4 - Создание каталога и файла

Добавьте текстовое содержимое в файл, например, строку приветствия (Рис. 5)

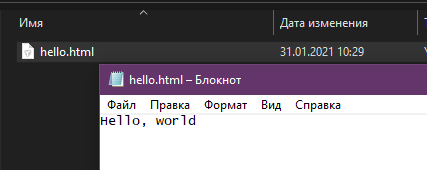


Рисунок 5 - Содержание файла

Инициализируйте репозиторий командой git init (Рис. 6)

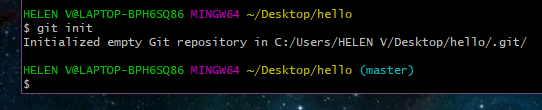


Рисунок 6 - Создание репозитория

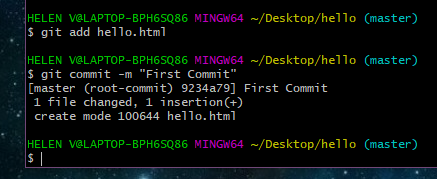
Зарегистрируйте файл в системе контроля версий через git add (Рис. 7)

Рисунок 7 - Добавление в репозиторий

**Мониторинг состояния** Текущий статус отслеживаемых файлов проверяется командой git status. При отсутствии изменений вывод соответствует Рис. 8.

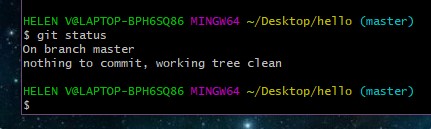


Рисунок 8 - Проверка состояния репозитория

**Внесение правок** Измените содержимое файла (Рис. 9)

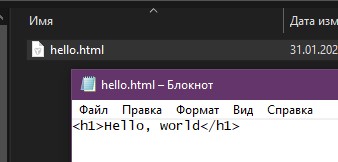


Рисунок 9 - Внесение изменений в файл

Повторная проверка статуса отобразит уведомление о незафиксированных изменениях (Рис. 10)

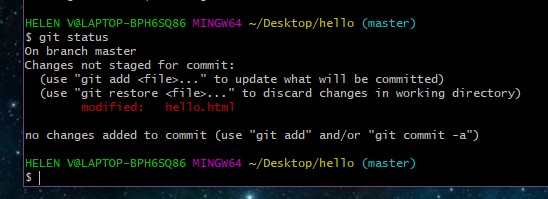


Рисунок 10 - Сообщение о незафиксированных изменениях

**Подготовка к фиксации**   
Для индексации изменений выполните последовательность команд из Рис. 11.

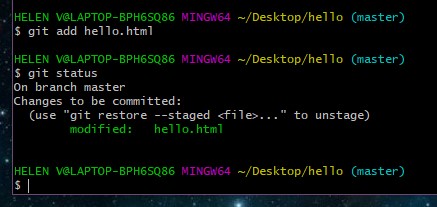


Рисунок 11 - Команды для индексации изменений

Индексацию можно отменить через git reset до момента фиксации.

**Многоэтапная фиксация** При работе с несколькими файлами возможна поэтапная фиксация:

* Создайте три файла (Рис. 12)
* Зафиксируйте первые два файла отдельным коммитом (Рис. 13)
* Третий файл добавьте следующим коммитом (Рис. 14)

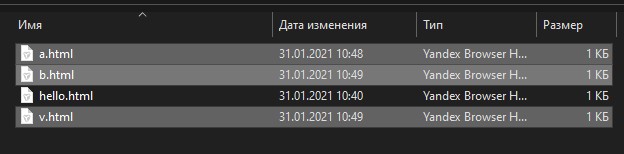


Рисунок 12 - Создано 3 файла

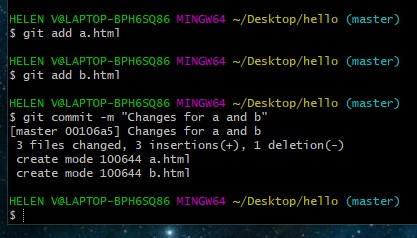


Рисунок 13 - Индексация и коммит для 2 файлов

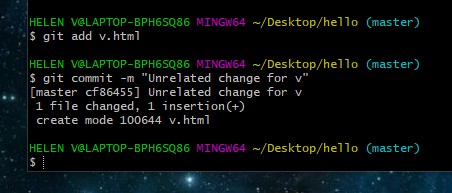


Рисунок 14 - Индексация и коммит для третьего файла

**Редактирование комментариев** Используйте git commit без параметра -m (Рис. 15)

1. В открывшемся редакторе укажите описание (Рис. 16)
2. После сохранения система выведет подтверждение (Рис. 17)
3. Проверьте статус репозитория (Рис. 18)

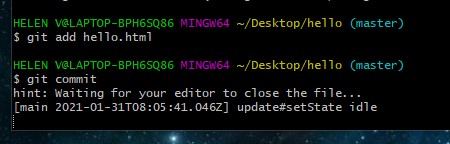


Рисунок 15 - Коммит изменений

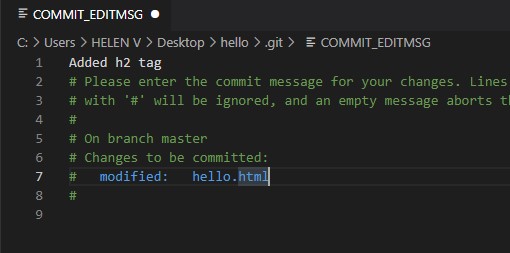


Рисунок 16 - Ввод комментария

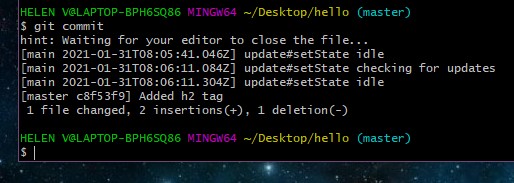


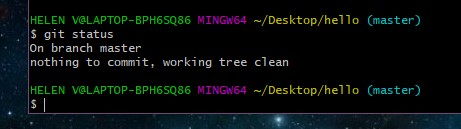
Рисунок 17 - Коммит-сообщения

Рисунок 18 - Проверка состояния

**Фиксация изменений вместо файлов** Система отслеживает изменения в файлах, а не сами файлы:

1. Добавьте HTML-теги в файл (Рис. 19) и проиндексируйте
2. Внесите дополнительные правки (тег head) без индексации (Рис. 20)
3. Проверка статуса покажет два состояния одного файла (Рис. 21)
4. Зафиксируйте первое изменение (Рис. 22)
5. Проиндексируйте и закоммитьте второе изменение (Рис. 23-24)

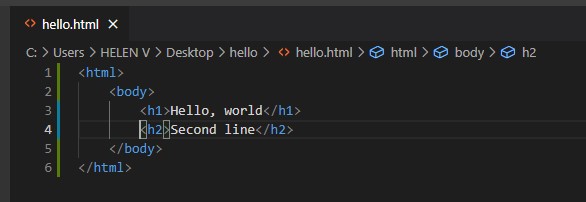


Рисунок 19 - Добавление тегов html и body

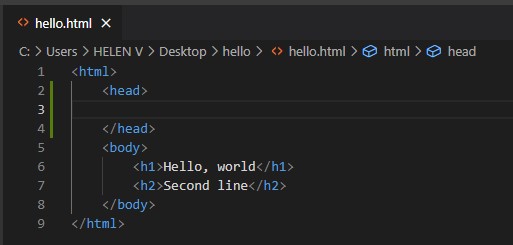


Рисунок 20 - Добавление тега head

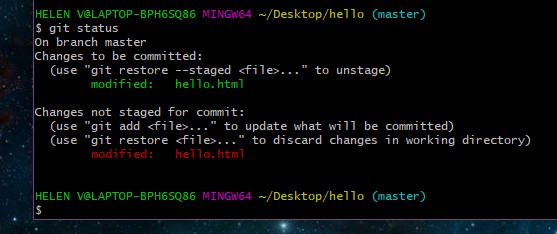


Рисунок 21 - Проверка состояния

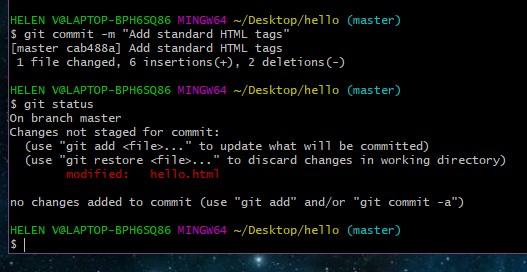


Рисунок 22 - Коммит 1 изменения и проверка состояния

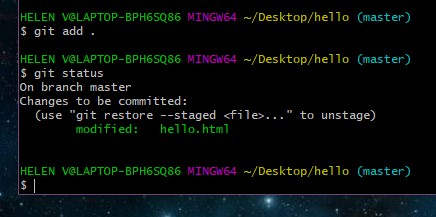


Рисунок 23 - Индексация 2 изменения и проверка состояния

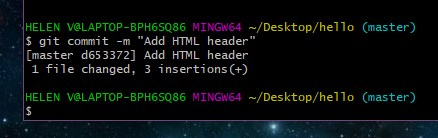
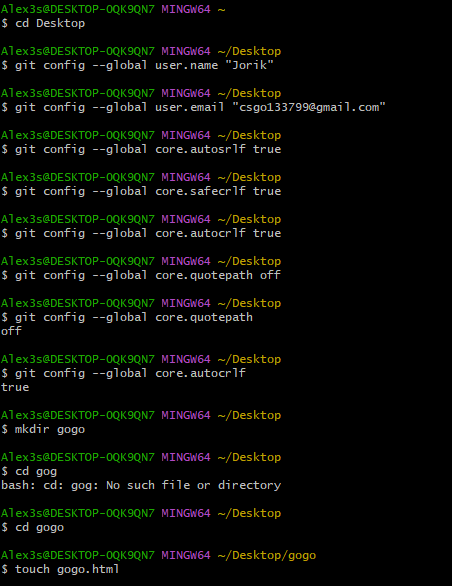


Рисунок 24 - Коммит 2 изменения



Изображение выглядит как текст, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.