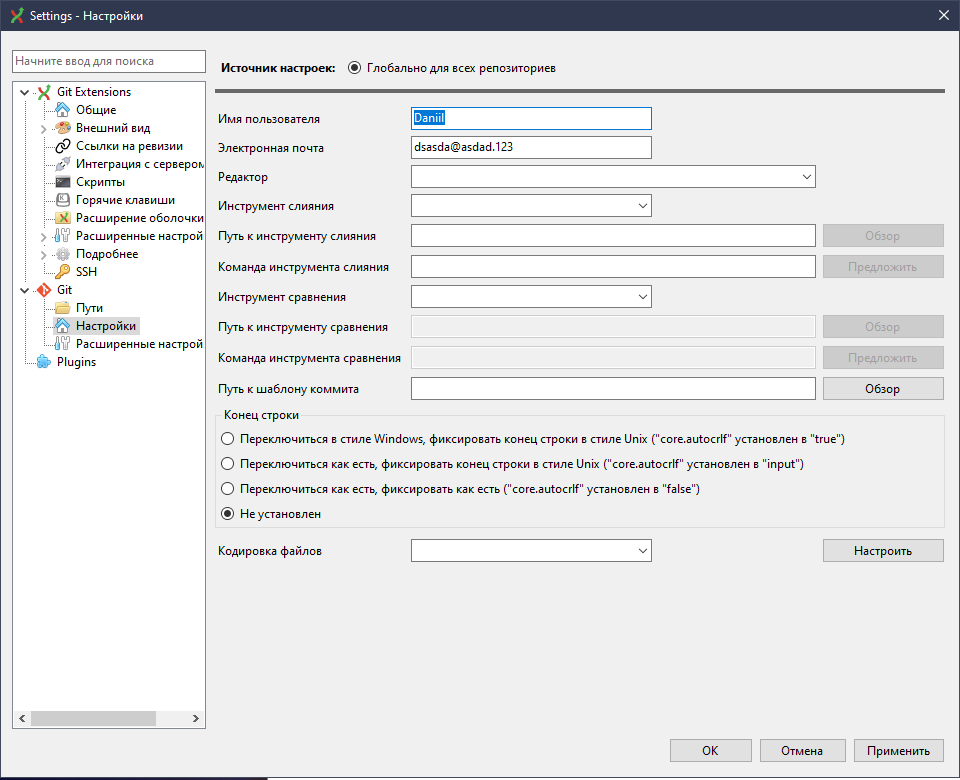
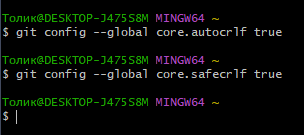
\_УП\_1

Если git никогда ранее не использовался пользователем, то необходимо установить имя и электронную почту. Для этого требуется выполнить команды, показанные на рисунке 1.



Затем нужно указать параметры установки окончаний строк (рисунок 2).



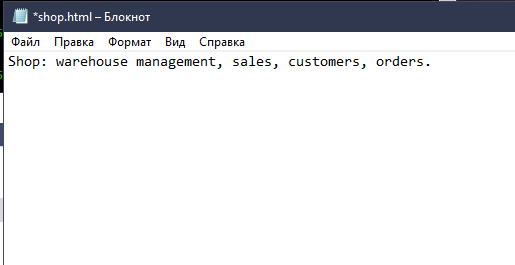
И последним пунктом идет установка отображения Unicode, показанная на рисунке 3.



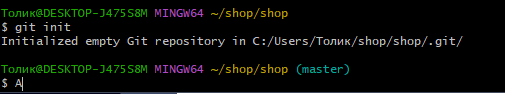
Сначала нужно создать пустой каталог и внутри него файл hello.html (рисунок 4).



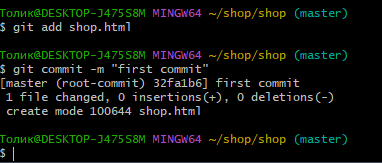
После этого в файл необходимо ввести данные, например, «Hello, world», как показано на рисунке 5.



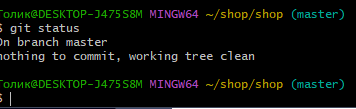
Для создания репозитория используется команда git init (рисунок 6).



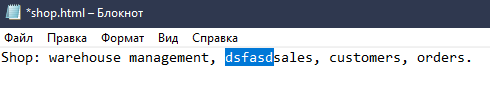
Для добавления страницы в репозиторий необходима команда git add (рисунок 7).



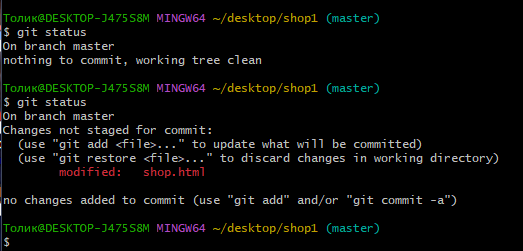
Проверка состояния репозитория осуществляется с помощью команды git status. Если в репозитории хранится текущее состояние рабочего каталога и нет изменений, ожидающих записи, будет показано сообщение, как на рисунке 8.



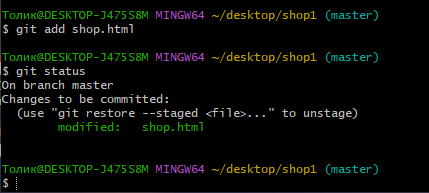
Сначала необходимо внести изменения в файл (рисунок 9).



Если после предыдущего пункта осуществить проверку состояния репозитория, то будет показано данное сообщение (рисунок 10).



Для того, чтобы проиндексировать изменения, нужно осуществить действия, показанные на рисунке 11.



После этого изменения файлы были проиндексированы. Это значит, что пока изменения не записаны в репозиторий. Если изменения позже не нужно будет фиксировать, то индексацию можно снять командой git reset.

Можно зафиксировать изменения отдельными коммитами. Как это сделать, показано на рисунках 12-14.

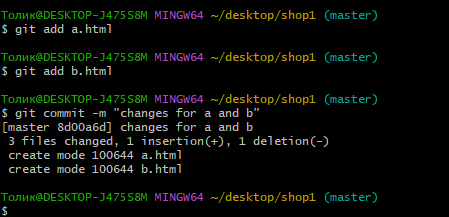


Рисунок 13 - Индексация и коммит для 2 файлов

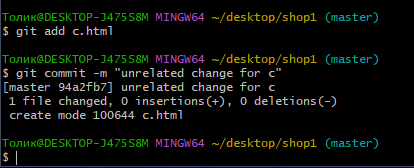


Рисунок 14 - Индексация и коммит для третьего файла

Для того, чтобы редактировать комментарий коммита, нужно использовать команду git commit без метки -m.

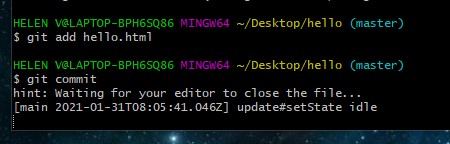


Рисунок 15 - Коммит изменений

После этого будет открыт редактор, в котором необходимо на 1 строке записать комментарий коммита (рисунок 16).

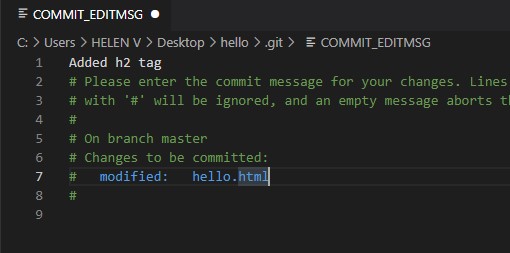


Рисунок 16 - Ввод комментария

После выхода из текстового редактора будет указано следующее сообщение (рисунок 17).

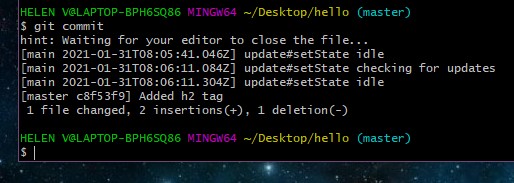


Рисунок 17 - Коммит-сообщения

После этого еще раз нужно проверить состояние репозитория (рисунок 18).

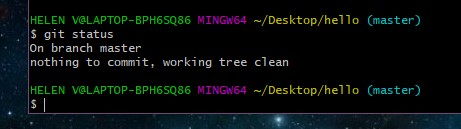


Рисунок 18 - Проверка состояния

## Изменения, а не файлы

Для того, чтобы понять, что git фокусируется на изменениях в файле, а не на самом файле, можно проделать следующие действия.

Сначала нужно в файл hello.html добавить теги html и body (рисунок 19), а затем проиндексировать изменения.

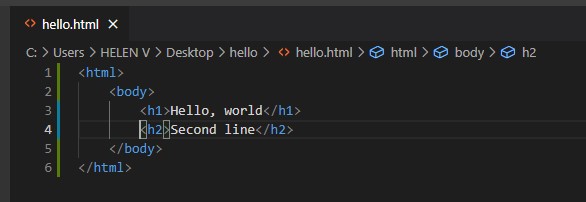


Рисунок 19 - Добавление тегов html и body

Затем еще раз нужно добавить изменения в файл (добавить тег head), но изменения не индексировать (рисунок 20).

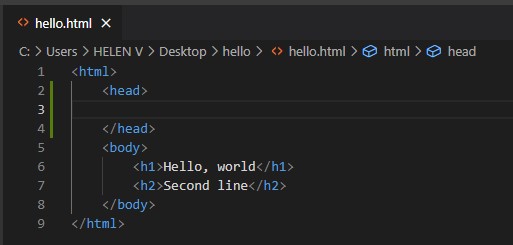


Рисунок 20 - Добавление тега head

Далее нужно проверить статус. На рисунке 21 видно, что файл hello.html указан дважды: первое изменение проиндексировано и готово к коммиту, а второе – нет.

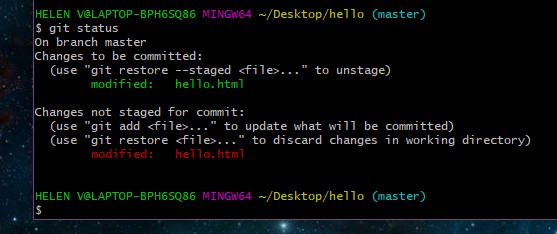


Рисунок 21 - Проверка состояния

Далее надо произвести коммит проиндексированного изменения и затем еще раз проверить состояние (рисунок 22).

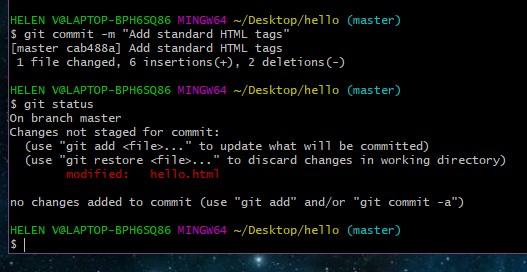


Рисунок 22 - Коммит 1 изменения и проверка состояния

Нужно добавить второе изменение в индекс и затем проверить состояние (рисунок 23).

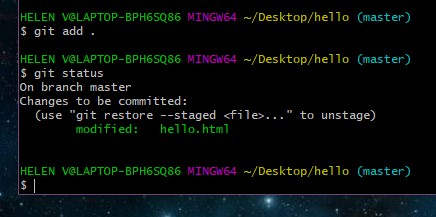


Рисунок 23 - Индексация 2 изменения и проверка состояния

После этого нужно сделать коммит второго изменения (рисунок 24).

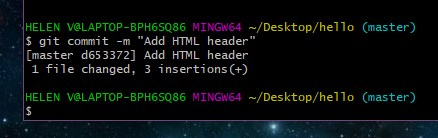


Рисунок 24 - Коммит 2 изменения