Git Guide Redo

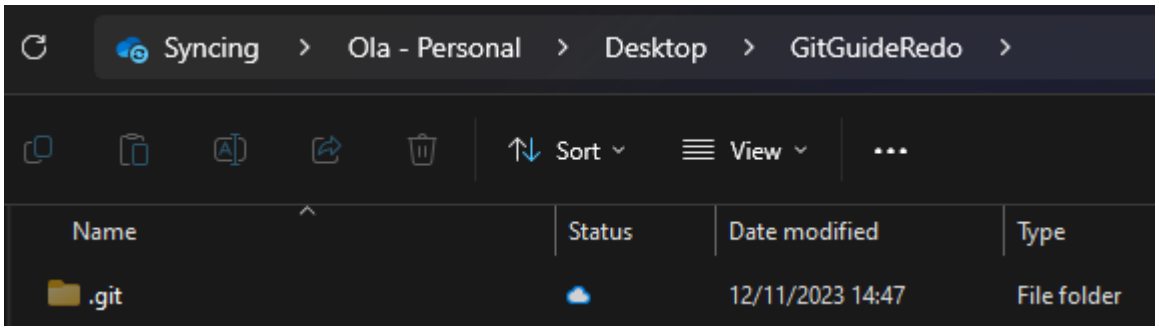
Aleksandra Skarzycka – 12/11/23

Git init

Git init creates a new empty repository in a chosen directory or initialize an existing directory as a git repository. When you run it, it sets up git's infrastructure and data structures to keep track of changes.

Syntax: git init

```
olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo
$ git init
Initialized empty Git repository in C:/Users/olask/OneDrive/Desktop/GitGuideRedo
/.git/
```

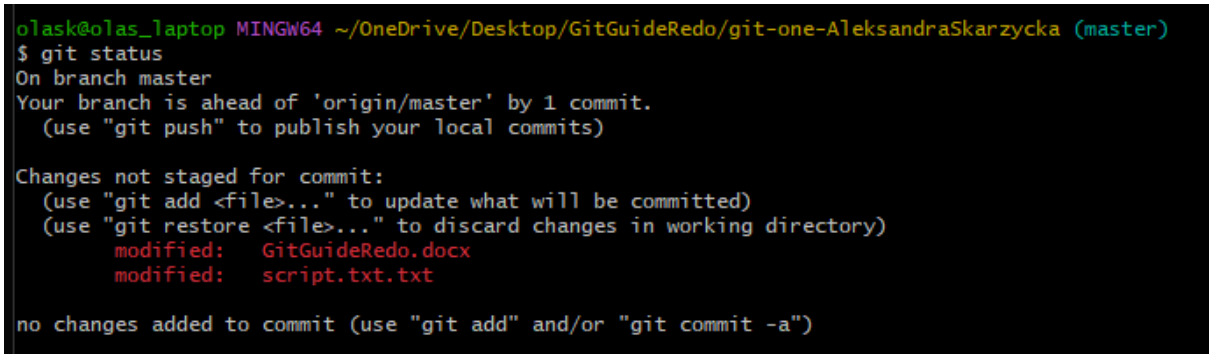


(This would be a hidden folder but if this shows up you know it worked)

Git status

Status displays information about the current state of the working directory and git repository. Shows which files have been modified and which ones are ready to be commit.

Syntax: git status



Git add

Git add command is used to add any staged changes in the working directory to be added into the next commit point.

Syntax: git add <filenamehere>

```
olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git add .

olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   GitGuideRedo.docx
        modified:   script.txt.txt
```

When you check the status again it should show the changes to be committed.

Git commit

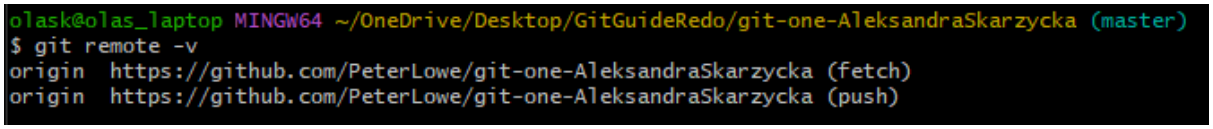
Git commit creates a snapshot of the staged changes made, saves your changes to a local repository. You must specifically tell git which files you want to include before you commit, or else it won't save them.

Syntax: git commit -m "your message here"

Git remote

Git remote lets you create, view, and delete connections to other repositories, accompanied by their URL. Remote connections are like bookmarks rather than direct links that let you access different repositories.

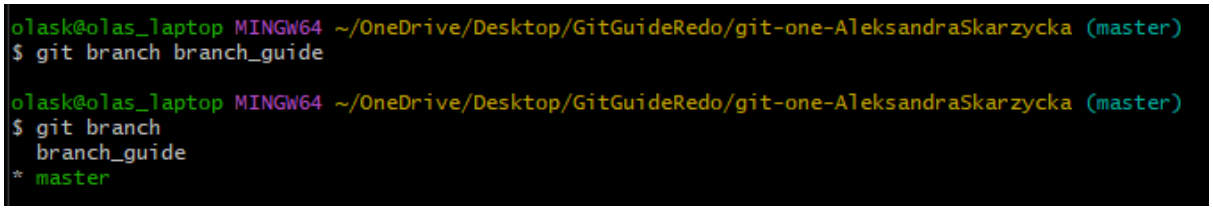
Syntax: git remote -v (this lists the repositories).



Git branch

Branches are used to list, create, delete, and manage branches in a git repository. Separate lines to work on different features.

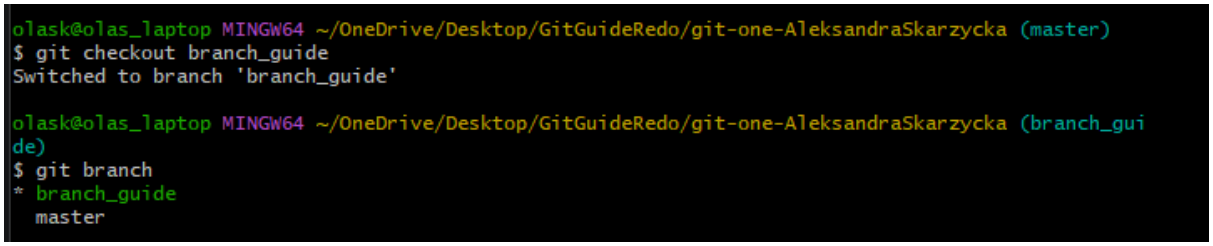
Syntax: git branch <branch_name> (creates branches), git branch (lists all branches)



Git checkout

Git checkout lets you navigate between branches created using git branch. Must commit your changes before you move between branches.

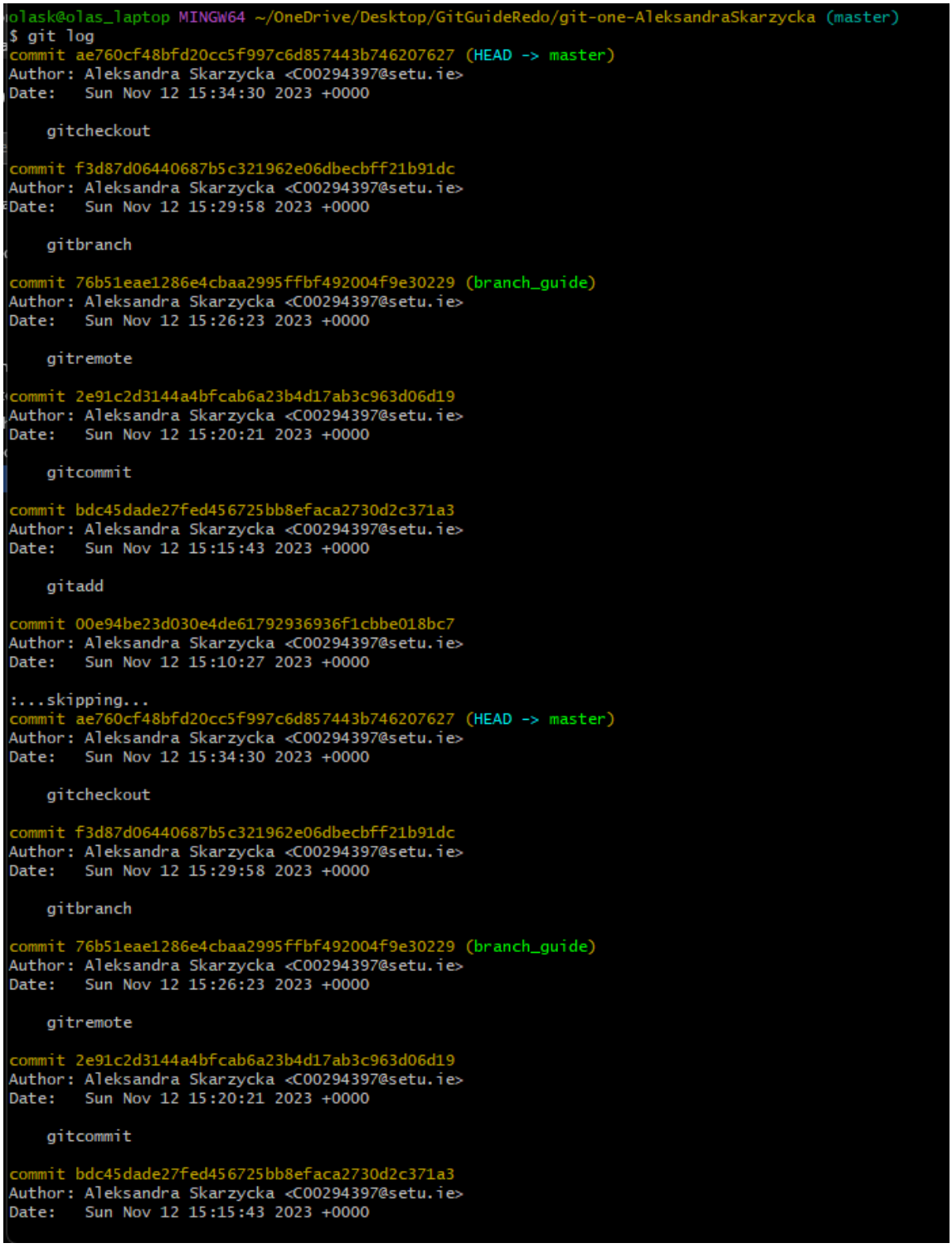
Syntax: git checkout <branch_name>



Git log

Displays the history of commits in a repository. The most recent commits show up first and it goes down the list with later ones. This command lists each commit with its SHA-1 checksum, the author's name and email, the date written, and the commit message.

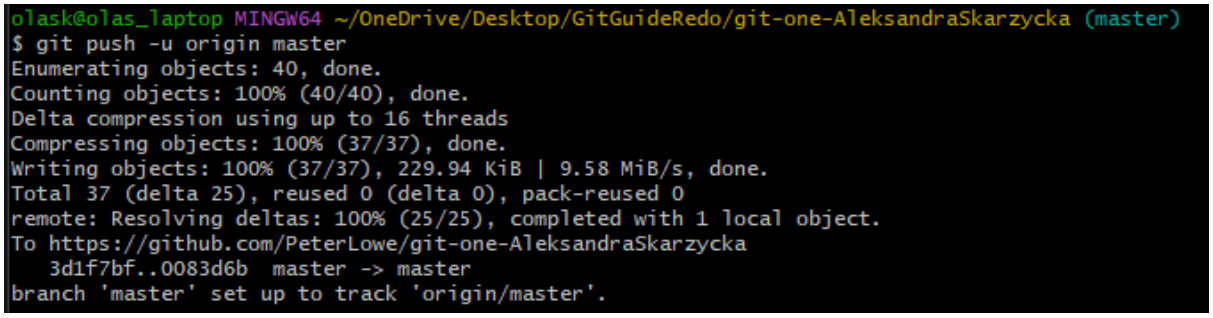
Syntax: git log



Git push

Git push allows you to upload your local repository changes to a remote repository. Allows you to share your work with the people you are collaborating with.

Syntax: git push -u origin master



Git stash

Lets you temporarily save your changes in your working directory that you don't want to commit yet. You use the command git stash and it saves it for you, use git stash list to see all the stashed saves.

Syntax: git stash (saves changes), git stash list (lists stashes)