

Git Guide Redo

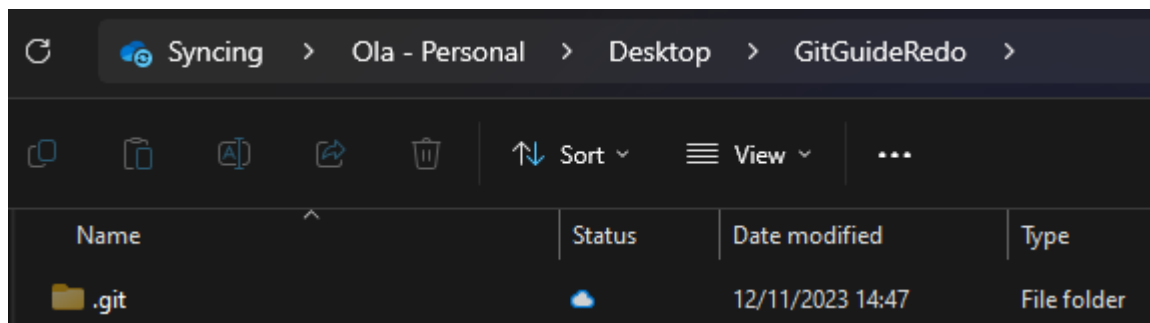
Aleksandra Skarzycka – 12/11/23

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

```
o1ask@o1as_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 not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   GitGuideRedo.docx
        modified:   script.txt.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

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>

```
o1ask@o1as_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git add .

o1ask@o1as_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"`

```
o1ask@o1as_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git commit -m gitcommit
[master 2e91c2d] gitcommit
2 files changed, 5 insertions(+)
```

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).

```
olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git remote -v
origin  https://github.com/PeterLowe/git-one-AleksandraSkarzycka (fetch)
origin  https://github.com/PeterLowe/git-one-AleksandraSkarzycka (push)
```

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)

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

olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git branch
  branch_guide
* master
```

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>

```
olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git checkout branch_guide
Switched to branch 'branch_guide'

olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (branch_guide)
$ git branch
* branch_guide
  master
```

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`

```
o1ask@o1as_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git log
commit ae760cf48bfd20cc5f997c6d857443b746207627 (HEAD -> master)
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:34:30 2023 +0000

    gitcheckout

commit f3d87d06440687b5c321962e06dbecbff21b91dc
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:29:58 2023 +0000

    gitbranch

commit 76b51eae1286e4cbaa2995ffbf492004f9e30229 (branch_guide)
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:26:23 2023 +0000

    gitremote

commit 2e91c2d3144a4bfcab6a23b4d17ab3c963d06d19
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:20:21 2023 +0000

    gitcommit

commit bdc45dade27fed456725bb8efaca2730d2c371a3
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:15:43 2023 +0000

    gitadd

commit 00e94be23d030e4de61792936936f1cbbe018bc7
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:10:27 2023 +0000

...skipping...
commit ae760cf48bfd20cc5f997c6d857443b746207627 (HEAD -> master)
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:34:30 2023 +0000

    gitcheckout

commit f3d87d06440687b5c321962e06dbecbff21b91dc
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:29:58 2023 +0000

    gitbranch

commit 76b51eae1286e4cbaa2995ffbf492004f9e30229 (branch_guide)
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:26:23 2023 +0000

    gitremote

commit 2e91c2d3144a4bfcab6a23b4d17ab3c963d06d19
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:20:21 2023 +0000

    gitcommit

commit bdc45dade27fed456725bb8efaca2730d2c371a3
Author: Aleksandra Skarzycka <C00294397@setu.ie>
Date: Sun Nov 12 15:15:43 2023 +0000
```


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`

```
olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git push -u origin master
Enumerating objects: 40, done.
Counting objects: 100% (40/40), done.
Delta compression using up to 16 threads
Compressing objects: 100% (37/37), done.
Writing objects: 100% (37/37), 229.94 KiB | 9.58 MiB/s, done.
Total 37 (delta 25), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (25/25), completed with 1 local object.
To https://github.com/PeterLowe/git-one-AleksandraSkarzycka
   3d1f7bf..0083d6b  master -> master
branch 'master' set up to track '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)

```
olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git stash
Saved working directory and index state WIP on master: 6069430 gitstash

olask@olas_laptop MINGW64 ~/OneDrive/Desktop/GitGuideRedo/git-one-AleksandraSkarzycka (master)
$ git stash list
stash@{0}: WIP on master: 6069430 gitstash
```