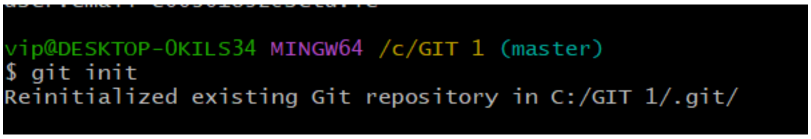
Git Lab Assignment 1

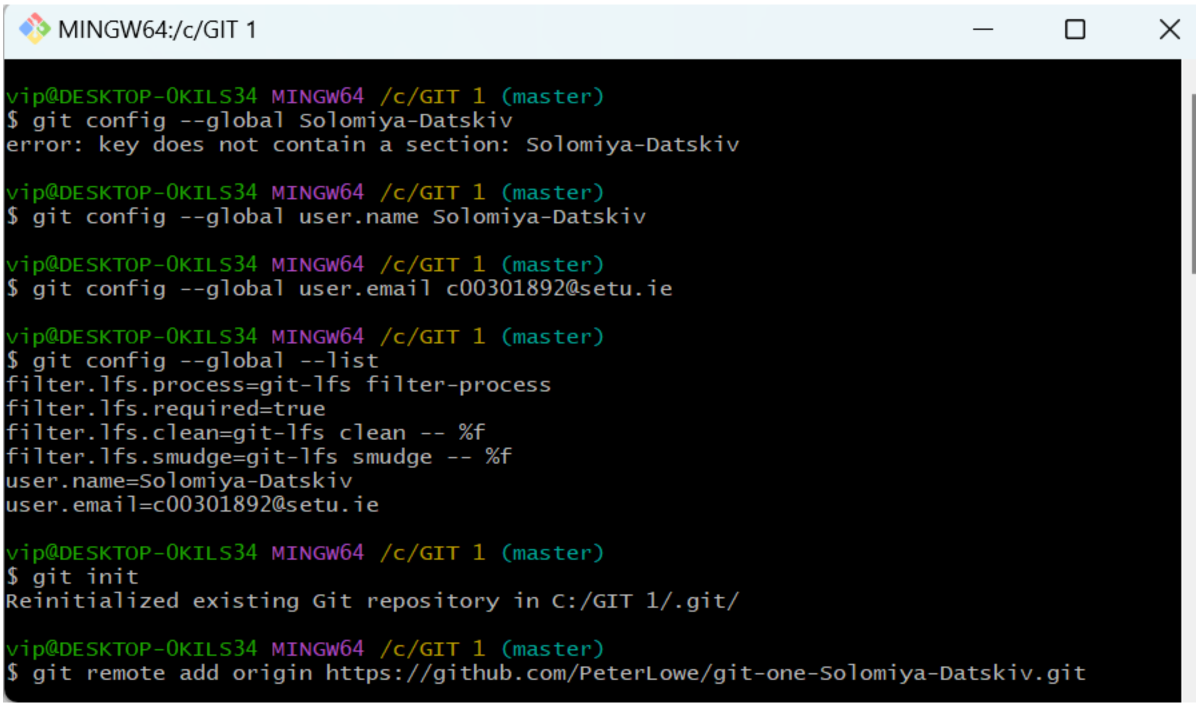
Solomiya Datskiv
C00301892
22/10/2023

1.Init

Command that allows user to create a new git repository.



```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git init
Reinitialized existing Git repository in C:/GIT 1/.git/
```



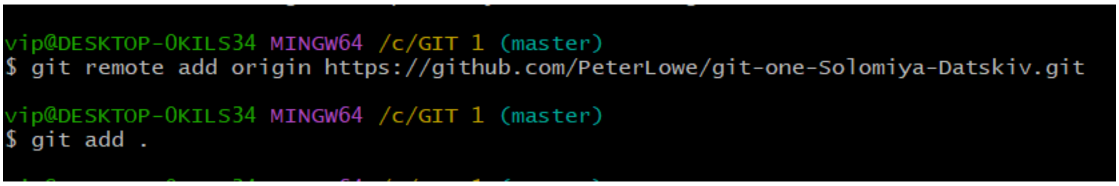
```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git config --global Solomiya-Datskiv
error: key does not contain a section: Solomiya-Datskiv

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git config --global user.name Solomiya-Datskiv

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git config --global user.email c00301892@setu.ie

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git config --global --list
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
user.name=Solomiya-Datskiv
user.email=c00301892@setu.ie

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git init
Reinitialized existing Git repository in C:/GIT 1/.git/

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git remote add origin https://github.com/PeterLowe/git-one-Solomiya-Datskiv.git
```

2. Add

Command allows to add new files to directory.

git add .

Adds files to next commit only.

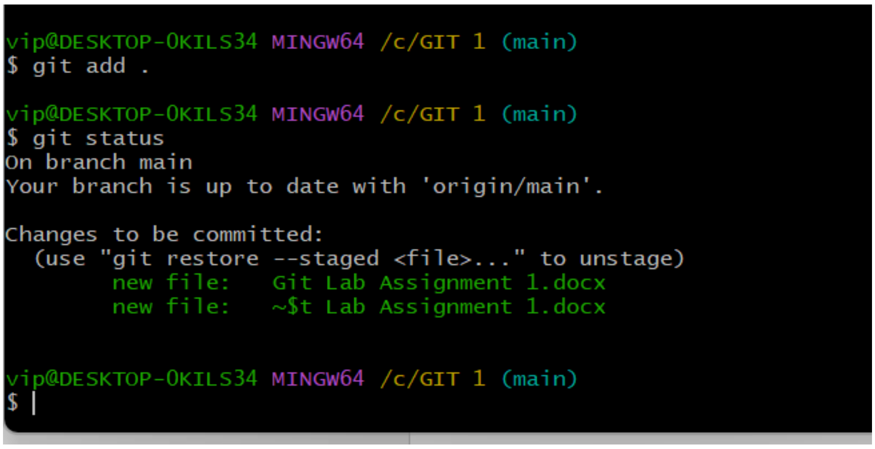


```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git remote add origin https://github.com/PeterLowe/git-one-Solomiya-Datskiv.git

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (master)
$ git add .
```

3. Status

git status

A command which lists files: modified file in working area, added files in staging area, current branch.



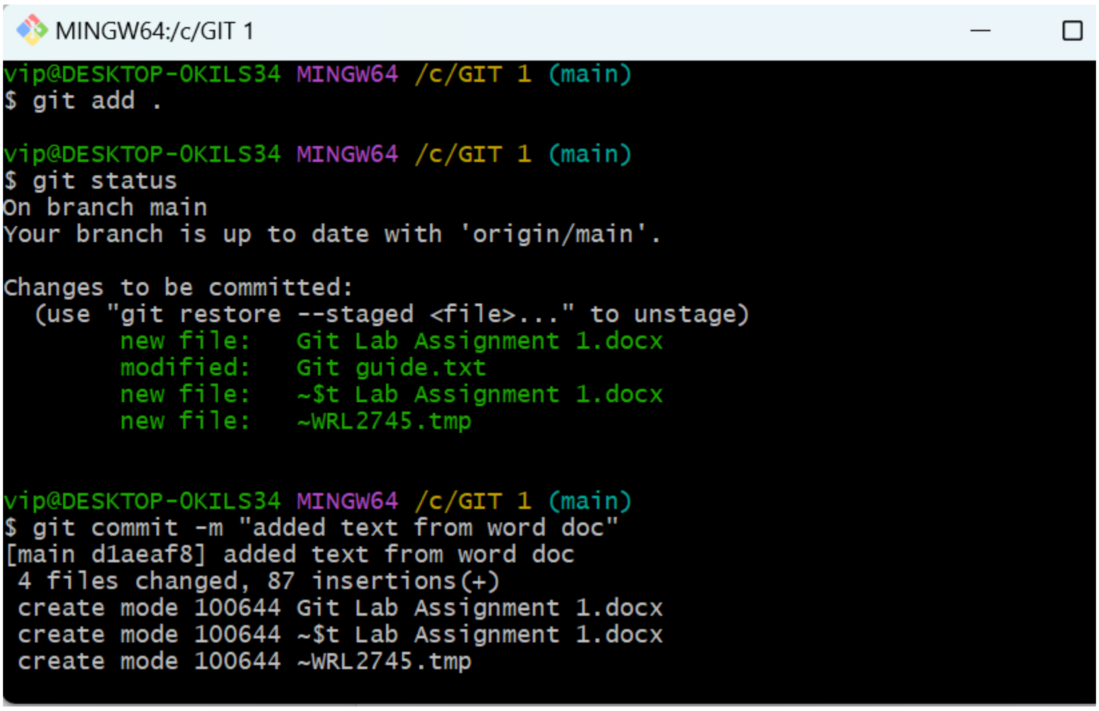
```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git add .

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   Git Lab Assignment 1.docx
        new file:   ~$t Lab Assignment 1.docx


vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ |
```

4. Commit

Git commit –m “comment”

Git commit

This command allows user to enter editor mode to enter multiline comment.



```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git add .

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   Git Lab Assignment 1.docx
        modified:   Git guide.txt
        new file:   ~$t Lab Assignment 1.docx
        new file:   ~WRL2745.tmp


vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git commit -m "added text from word doc"
[main d1aeaf8] added text from word doc
 4 files changed, 87 insertions(+)
 create mode 100644 Git Lab Assignment 1.docx
 create mode 100644 ~$t Lab Assignment 1.docx
 create mode 100644 ~WRL2745.tmp
```

5. Remote

Git remote –v

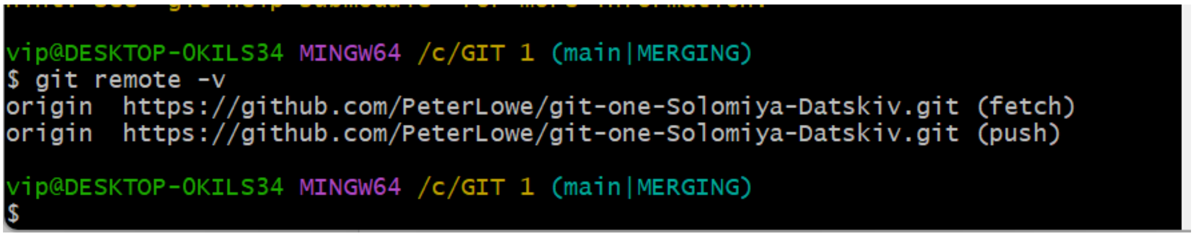
This command lists repositories on remote server.

Git remote add <name> <url>

Command creates a new remote repository.

Git remote rm <name>

Command deletes a remote repository.



```
hint: See 'git help submodule' for more information.

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$ git remote -v
origin  https://github.com/PeterLowe/git-one-Solomiya-Datskiv.git (fetch)
origin  https://github.com/PeterLowe/git-one-Solomiya-Datskiv.git (push)

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$
```

6. Push

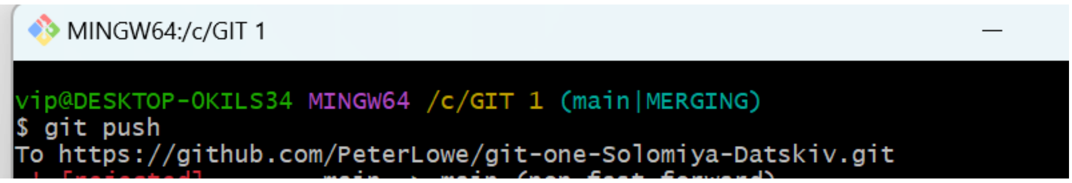
Git push

The command allows user to update remote node with current node.

Needs to associate remote node.

git push –u origin master

The command allows to push local master to origin on server.



```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$ git push
To https://github.com/PeterLowe/git-one-Solomiya-Datskiv.git
 ! [rejected]        main -> main (non-fast-forward)
```

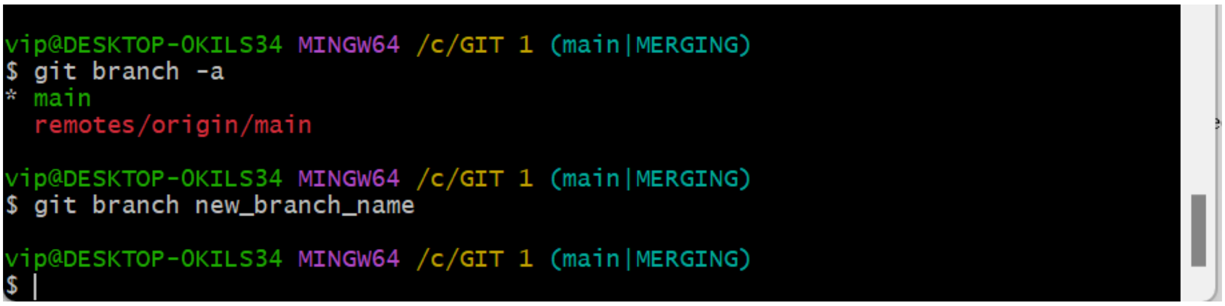
7. Branch

To create a new branch in the local repository, you have to use

Git branch new_branch_name

To list all branches in repository:

Git branch –a/-l/-r



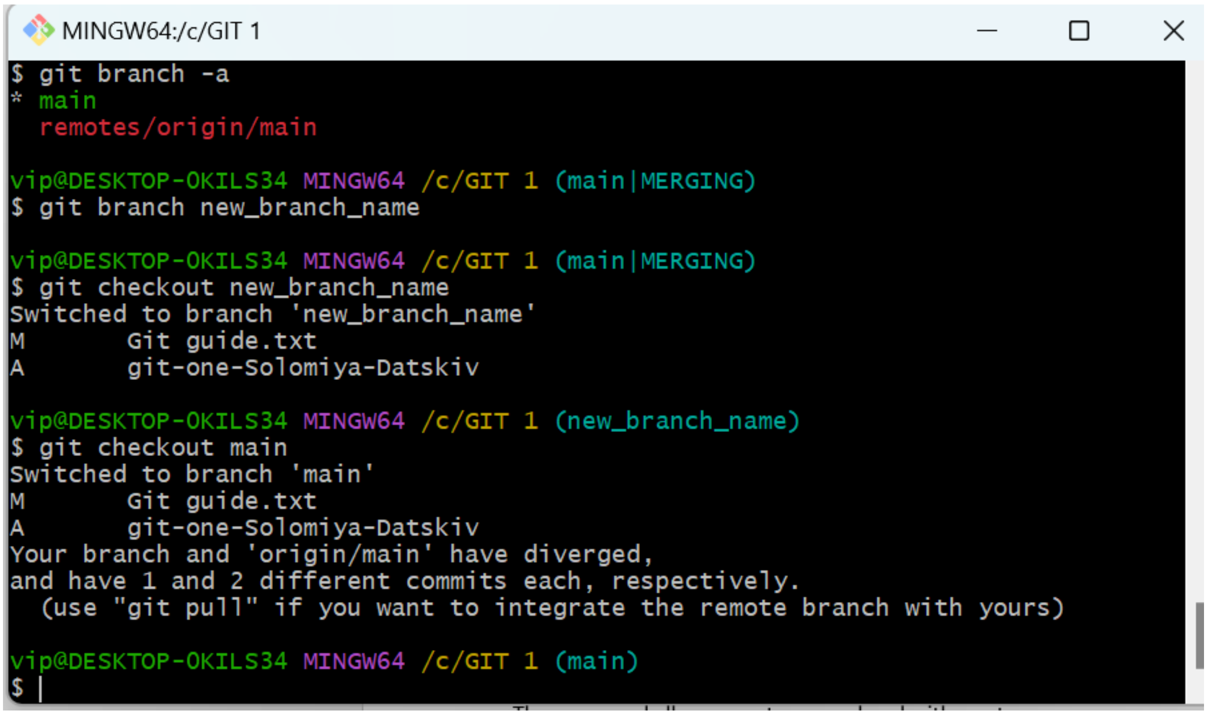
```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$ git branch -a
* main
  remotes/origin/main

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$ git branch new_branch_name

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$ |
```

8. Checkout

Git checkout branch_name

The command makes this node the current HEAD and ensures working directory matches that node / commit.

It won’t work if there are modified files.



```
$ git branch -a
* main
  remotes/origin/main

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$ git branch new_branch_name

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$ git checkout new_branch_name
Switched to branch 'new_branch_name'
M       Git guide.txt
A       git-one-Solomiya-Datskiv

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (new_branch_name)
$ git checkout main
Switched to branch 'main'
M       Git guide.txt
A       git-one-Solomiya-Datskiv
Your branch and 'origin/main' have diverged,
and have 1 and 2 different commits each, respectively.
  (use "git pull" if you want to integrate the remote branch with yours)

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ |
```

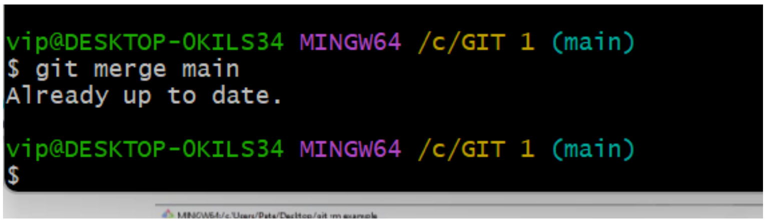
9. Merge

git merge master

The command allows user to merge head with master.

! - Git will automatically apply changes if separate.

You should edit file/s and add first, and then commit.



```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git merge main
Already up to date.

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$
```

10. Log

The command displays entire commit history:

Git log

To alter files and change line:

Git log --stat

Branch paths drawn:

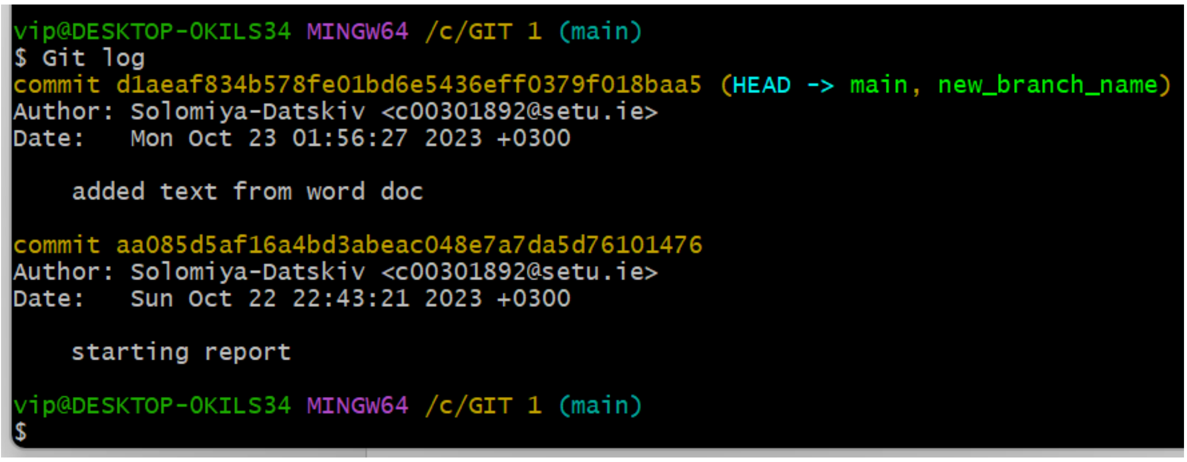
Git log --graph

To draw paths condensed output:

Git log --graph --oneline

Only last n commits:

Git log -n



```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ Git log
commit d1aeaf834b578fe01bd6e5436eff0379f018baa5 (HEAD -> main, new_branch_name)
Author: Solomiya-Datskiv <c00301892@setu.ie>
Date:   Mon Oct 23 01:56:27 2023 +0300

    added text from word doc

commit aa085d5af16a4bd3abeac048e7a7da5d76101476
Author: Solomiya-Datskiv <c00301892@setu.ie>
Date:   Sun Oct 22 22:43:21 2023 +0300

    starting report

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$
```

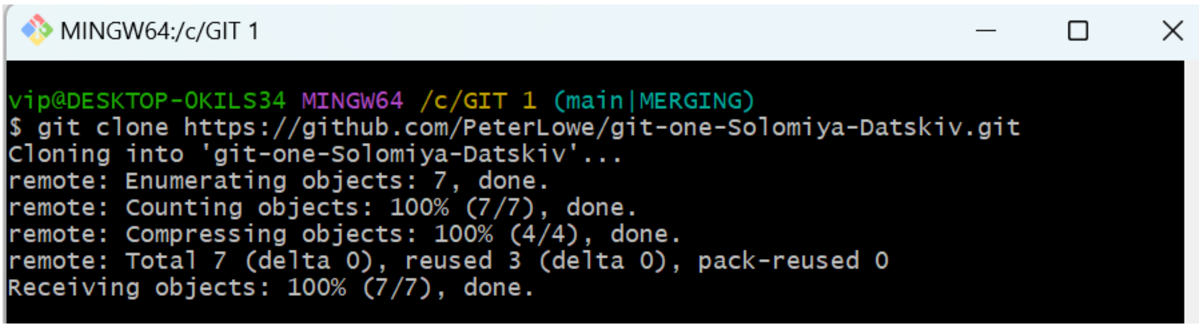
11. Clone

Git clone URL

This command allows to make a copy of a repository that currently exists on GitHub (or other) and save it on PC. After executing clone the repo will be in a sub folder of where user ran the git clone command.

To access the repo:

Cd DirectoryName



```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main|MERGING)
$ git clone https://github.com/PeterLowe/git-one-Solomiya-Datskiv.git
Cloning into 'git-one-Solomiya-Datskiv'...
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 7 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (7/7), done.
```

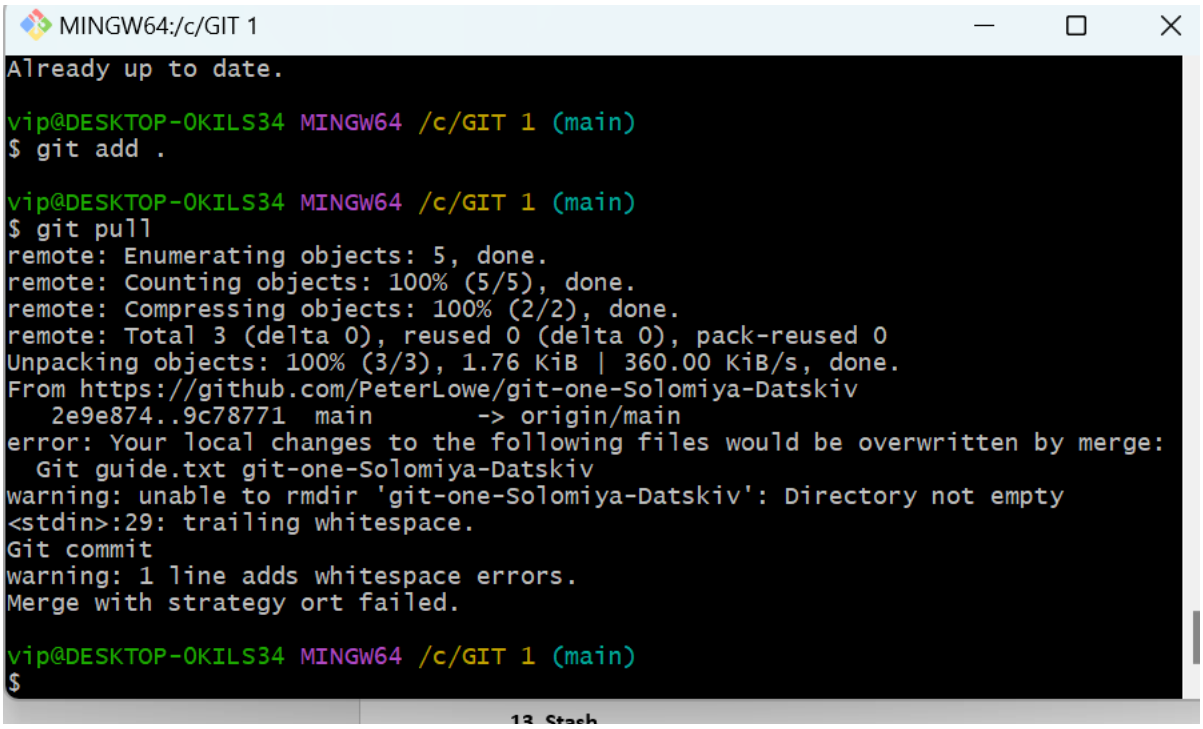
12. Pull

Git pull <remote>

The command performs a fetch and merge.

Git pull –rebase

Performs a fetch and merges the remote origin into local master.



```
Already up to date.

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git add .

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 1.76 KiB | 360.00 KiB/s, done.
From https://github.com/PeterLowe/git-one-Solomiya-Datskiv
   2e9e874..9c78771  main       -> origin/main
error: Your local changes to the following files would be overwritten by merge:
  Git guide.txt git-one-Solomiya-Datskiv
warning: unable to rmdir 'git-one-Solomiya-Datskiv': Directory not empty
<stdin>:29: trailing whitespace.
Git commit
warning: 1 line adds whitespace errors.
Merge with strategy ort failed.

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$
```

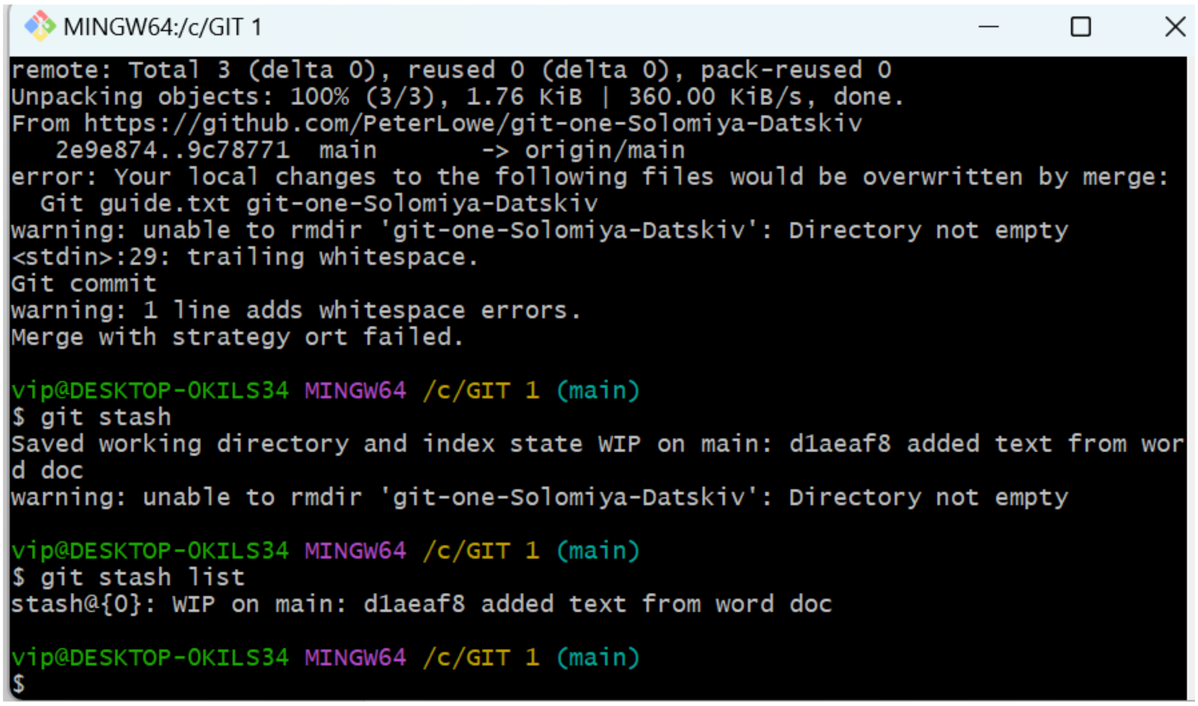
13. Stash

Git stash

The command allows user to create a local only copy of the current working directory.

Git stash list

Allows user to list previously pushed stash(es).



```
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 1.76 KiB | 360.00 KiB/s, done.
From https://github.com/PeterLowe/git-one-Solomiya-Datskiv
   2e9e874..9c78771  main       -> origin/main
error: Your local changes to the following files would be overwritten by merge:
  Git guide.txt git-one-Solomiya-Datskiv
warning: unable to rmdir 'git-one-Solomiya-Datskiv': Directory not empty
<stdin>:29: trailing whitespace.
Git commit
warning: 1 line adds whitespace errors.
Merge with strategy ort failed.

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git stash
Saved working directory and index state WIP on main: d1aeaf8 added text from wor
d doc
warning: unable to rmdir 'git-one-Solomiya-Datskiv': Directory not empty

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git stash list
stash@{0}: WIP on main: d1aeaf8 added text from word doc

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$
```

14. Rm

Command which allows user to remove file from tracking:

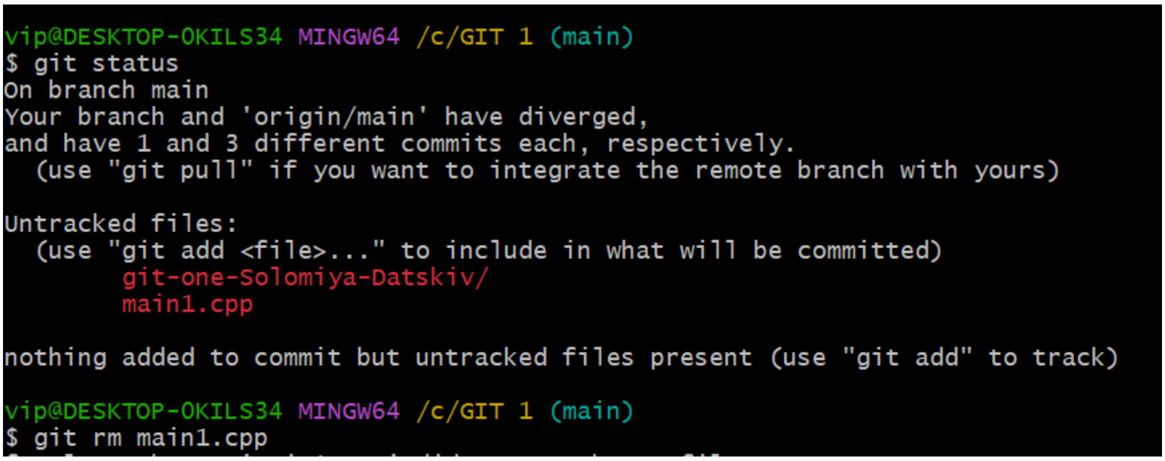
Git rm <filename>

Also:

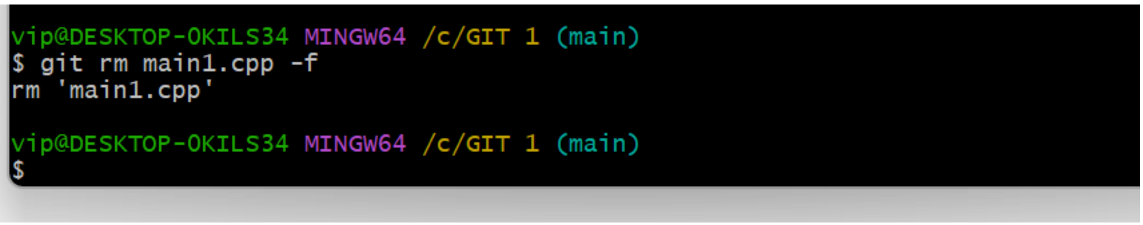
-- cached (staging area only)

-r (recursive)

-n (dry run list files only)



```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git status
On branch main
Your branch and 'origin/main' have diverged,
and have 1 and 3 different commits each, respectively.
  (use "git pull" if you want to integrate the remote branch with yours)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        git-one-Solomiya-Datskiv/
        main1.cpp

nothing added to commit but untracked files present (use "git add" to track)

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git rm main1.cpp
```



```
vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$ git rm main1.cpp -f
rm 'main1.cpp'

vip@DESKTOP-OKILS34 MINGW64 /c/GIT 1 (main)
$
```