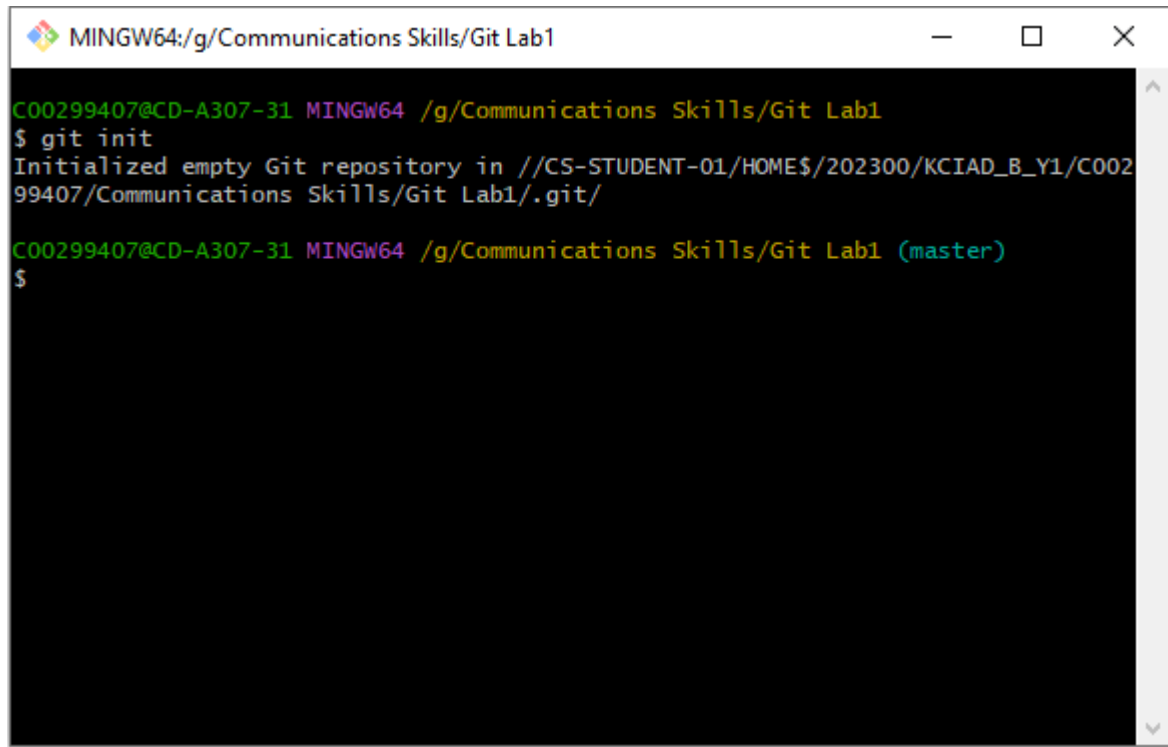


Init

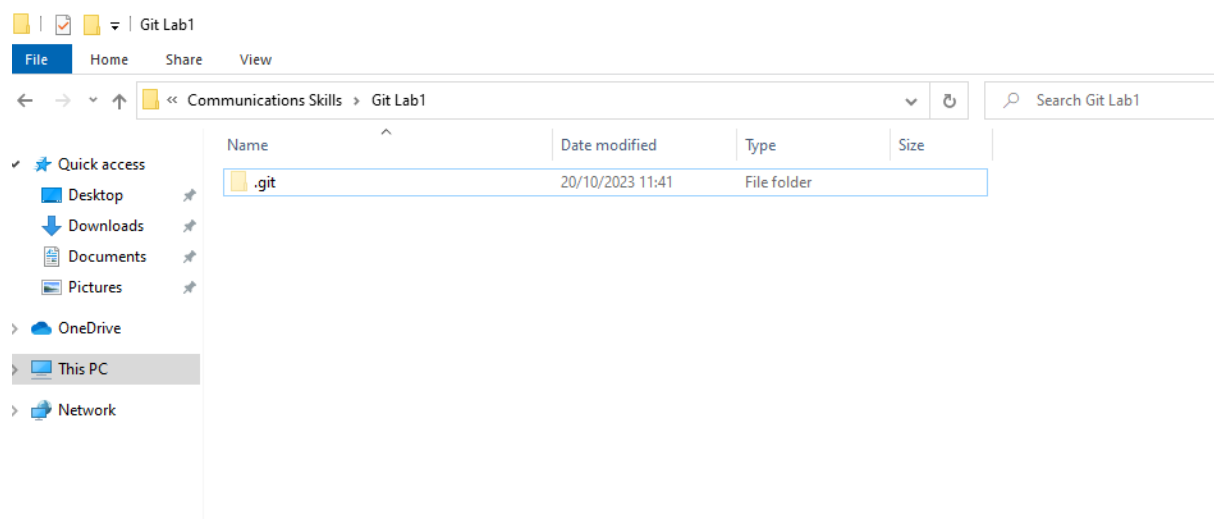
- Creates a new local repository in a chosen folder.
- Git Bash adds a .git folder to the chosen folder.
- This git bash folder is hidden but can be made visible through clicking “view” and selecting “hidden items”.



```
MINGW64:/g/Communications Skills/Git Lab1

C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1
$ git init
Initialized empty Git repository in //CS-STUDENT-01/HOME$/202300/KCIAD_B_Y1/C00299407/Communications Skills/Git Lab1/.git/

C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (master)
$
```



Add

- Adds new files to the directory.
- Doesn't include files ending with ".gitignore".
- Adds files to the next commit only.
- If a file is changed before committing, it needs to be added again.

The screenshot shows a terminal window with the command `git add Script1.txt` being executed. Below the terminal, a file explorer window displays the contents of the 'Git Lab1' directory. The file explorer shows a table with columns: Name, Date modified, Type, and Size. The files listed are: .git (File folder, 20/10/2023 12:17), Git report (Microsoft Word Document, 87 KB, 20/10/2023 12:15), README (Markdown Source File, 1 KB, 20/10/2023 12:16), and Script1 (Text Document, 1 KB, 20/10/2023 12:00). The terminal output shows the command being executed and the file being added to the staging area.

Name	Date modified	Type	Size
.git	20/10/2023 12:17	File folder	
Git report	20/10/2023 12:15	Microsoft Word Document	87 KB
README	20/10/2023 12:16	Markdown Source File	1 KB
Script1	20/10/2023 12:00	Text Document	1 KB

Status

- Gives a list of files that are modified in the working area.
- Gives a list of files that are added to the staged area.
- Gives a list of files that are in the current branch.

The screenshot shows a terminal window with the command `git status` being executed. The output shows the current branch as 'main' and lists the changes to be committed. The changes are: new file: Git report.docx, new file: README.md, and new file: ~\$t report.docx. The terminal output also includes instructions on how to unstage files using `git restore --staged <file>...`.

```
(use "git restore --staged <file>..." to unstage)
new file:  README.md

Untracked files:
(use "git add <file>..." to include in what will be committed)
Git report.docx
~$t report.docx

C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git add .

C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   Git report.docx
        new file:   README.md
        new file:   ~$t report.docx

C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (main)
$
```

Commit

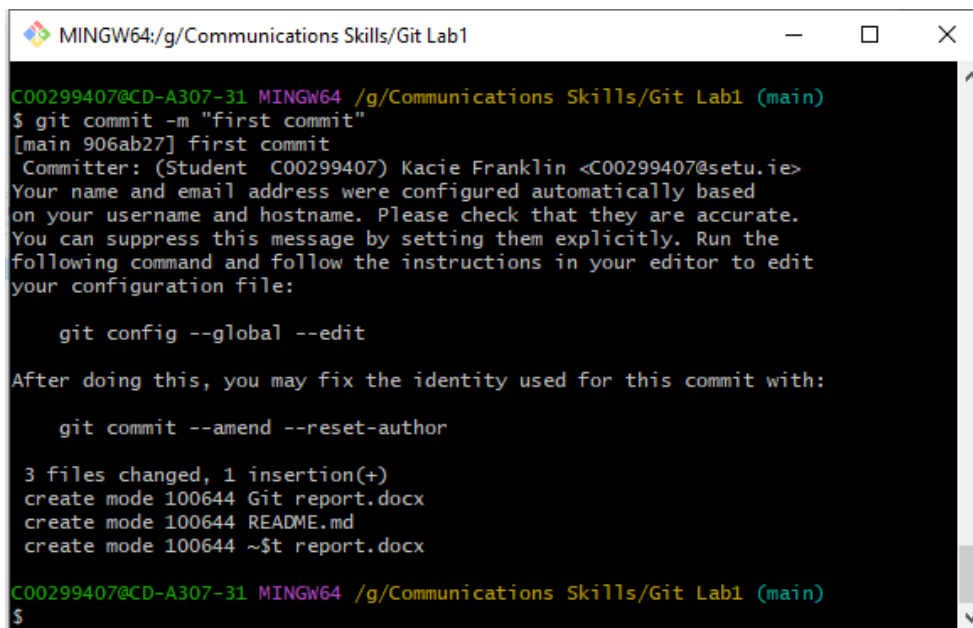
- Creates a snapshot of the current repository.

`git commit -m "comment"`

- Whenever a file is modified; a commit command needs to be done to have the latest version of the file saved to the repository.

`git commit --amend`

- This allows for this commit to be joined to previous commits instead of it being a separate one.



```
MINGW64:/g/Communications Skills/Git Lab1
C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git commit -m "first commit"
[main 906ab27] first commit
Committer: (Student C00299407) Kacie Franklin <C00299407@setu.ie>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

3 files changed, 1 insertion(+)
create mode 100644 Git report.docx
create mode 100644 README.md
create mode 100644 ~$t report.docx
C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (main)
$
```

`Push`

- Updates the remote node with current node.

`git push -u origin master`

- The local master is connected with the remote master.
- This command is only needed once.

```

MINGW64:/g/Communications Skills/Git Lab1
create mode 100644 Git report.docx
create mode 100644 README.md
create mode 100644 ~$t report.docx

C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git remote add origin https://github.com/PeterLowe/git-one-KacieFranklin.git
error: remote origin already exists.

C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 16 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (8/8), 78.44 KiB | 9.80 MiB/s, done.
Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/PeterLowe/git-one-KacieFranklin.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

C00299407@CD-A307-31 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ |

```

Branch

git branch new branch name

- Makes a new branch in the chosen local repository.

git branch -a/-l/-r

- Lists all branches in local repository

```

MINGW64:/g/Communications Skills/Git Lab1
$ git add .

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (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:   git2.txt

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git merge master
merge: master - not something we can merge

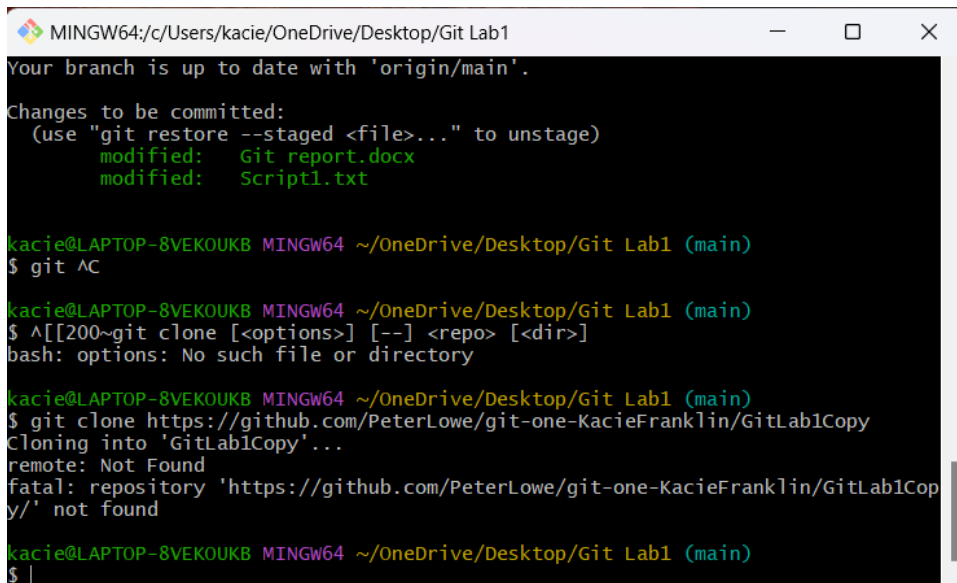
C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git branch -a
* main
  new_branch_name
  remotes/origin/main

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ |

```

Clone

- Creates a copy of the code from the repository which can then be worked on locally. Updates to the file can be sent to the repository when completed or at least updated often. Updates can also be received from others via the server. (GitHub)



```
MINGW64:/c/Users/kacie/OneDrive/Desktop/Git Lab1
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   Git report.docx
        modified:   Script1.txt

kacie@LAPTOP-8VEKOUKB MINGW64 ~/OneDrive/Desktop/Git Lab1 (main)
$ git ^C

kacie@LAPTOP-8VEKOUKB MINGW64 ~/OneDrive/Desktop/Git Lab1 (main)
$ ^[[200~git clone [<options>] [--] <repo> [<dir>]
bash: options: No such file or directory

kacie@LAPTOP-8VEKOUKB MINGW64 ~/OneDrive/Desktop/Git Lab1 (main)
$ git clone https://github.com/PeterLowe/git-one-KacieFranklin/GitLab1Copy
Cloning into 'GitLab1Copy'...
remote: Not Found
fatal: repository 'https://github.com/PeterLowe/git-one-KacieFranklin/GitLab1Copy/' not found

kacie@LAPTOP-8VEKOUKB MINGW64 ~/OneDrive/Desktop/Git Lab1 (main)
$ |
```

Pull

git pull <remote>

- Does a fetch command and a merge command all in one.

git pull -rebase

- Does a fetch and merge command while pulling from a remote origin and placing the files into the local master.
- (I attempted to send files to my own repository, but it saved them to the remote one instead of my own repository I had made and I don't know how to fix it)

```
MINGW64:/g/Communications Skills/Git Lab1

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git push -u origin main
info: please complete authentication in your browser...
Everything up-to-date
branch 'main' set up to track 'origin/main'.

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git push -u origin main
Everything up-to-date
branch 'main' set up to track 'origin/main'.

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git commit -m "commit 3"
[main fd4cf75] commit 3
Committer: (Student C00299407) Kacie Franklin <C00299407@setu.ie>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit
```

Merge

git merge master

- Merges the head with the master branch.
- Will automatically update if they are separate.
- Will warn if there's any issues with the merge and cancel the merge.
- Edit files, add and then commit.

Rm

- Does the opposite of the add command.
- Removes files from the directory.
- -- cached, removes files from the staging area only.
- -r, removes everything from the directory.

```
MINGW64:/g/Communications Skills/Git Lab1

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (master)
$ git status
On branch master
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   git2.txt

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (master)
$ git merge master
Already up to date.

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (master)
$ git branch -m main

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (main)
$ git merge master
merge: master - not something we can merge

C00299407@CD-A003-13 MINGW64 /g/Communications Skills/Git Lab1 (main)
$
```

- -n, removes dry run list files only.

```
MINGW64~/c:/Users/kacie/OneDrive/Desktop/Git Lab1
unknown option: -f
usage: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
          [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
          [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
          [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
          [--config-env=<name>=<envvar>] <command> [<args>]

kacie@LAPTOP-8VEKOUKB MINGW64 ~/OneDrive/Desktop/Git Lab1 (main)
$ git rm 'To be deleted.txt'
error: the following file has changes staged in the index:
    To be deleted.txt
(use --cached to keep the file, or -f to force removal)

kacie@LAPTOP-8VEKOUKB MINGW64 ~/OneDrive/Desktop/Git Lab1 (main)
$ git -f
unknown option: -f
usage: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
          [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
          [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
          [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
          [--config-env=<name>=<envvar>] <command> [<args>]

kacie@LAPTOP-8VEKOUKB MINGW64 ~/OneDrive/Desktop/Git Lab1 (main)
$ |
```