

Exercice1

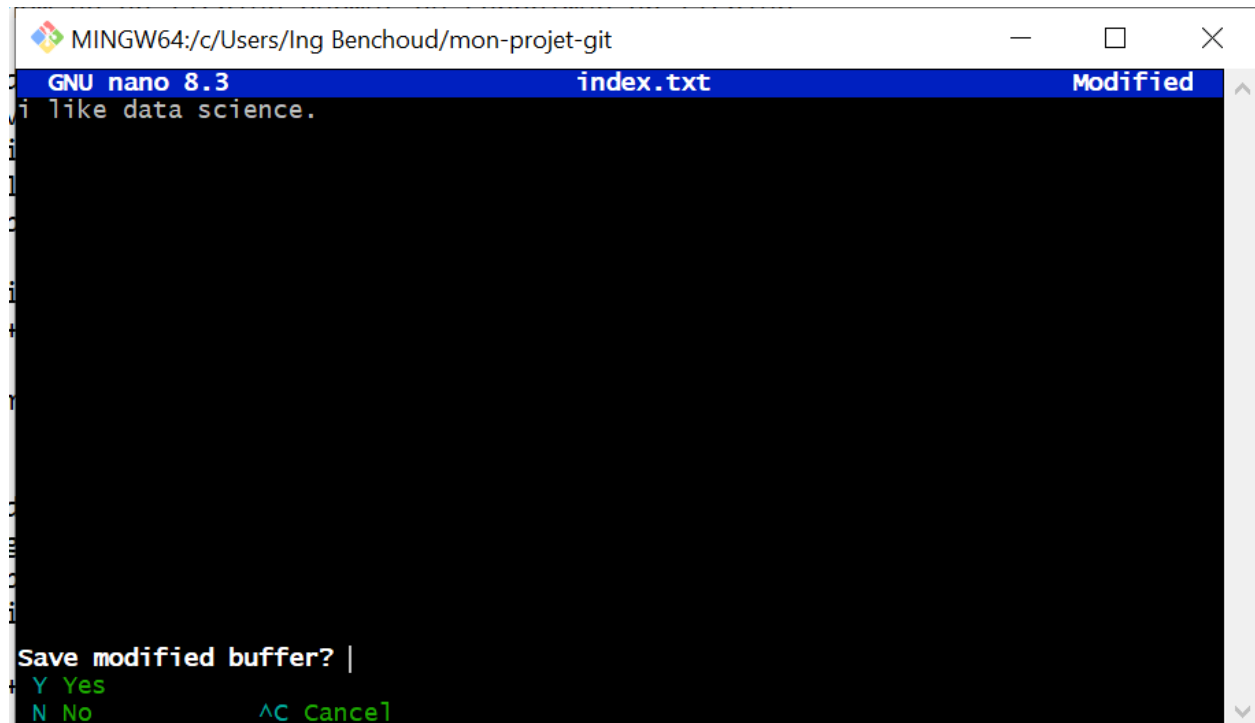
1. Créer un dossier mon-projet-git

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~ (master)
$ mkdir mon-projet-git
```

2. Créer un fichier index.txt avec une phrase

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (master)
$ touch index.txt
```

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (master)
$ nano index.txt
```



The screenshot shows a terminal window with the title 'MINGW64:/c/Users/Ing Benchoud/mon-projet-git'. Inside, the nano 8.3 text editor is open, editing 'index.txt'. The text 'i like data science.' is visible. At the bottom, a prompt asks 'Save modified buffer? |' with options 'Y Yes', 'N No', and '^C Cancel'.

3. Initialiser Git et fais un premier commit

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~ (master)
$ cd mon-projet-git

Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (master)
$ git add index.txt
warning: in the working copy of 'mon-projet-git/index.txt', LF will be replaced by CRLF the next time Git touches it

Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (master)
$ git commit -m "premiere modification" -a
[master (root-commit) b2f95a2] premiere modification
1 file changed, 2 insertions(+)
create mode 100644 mon-projet-git/index.txt
```

Exercice 2

1. Modifier index.txt

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (master)
$ nano index.txt
```

```
GNU nano 8.3 index.txt Modified
I like data science very much.
```

AG Help AO Write Out AF Where Is AK Cut AT Execute AC Location M-U Undo M-A Set Mark M-J To Bracket M-B Previous
 AX Exit AR Read File AL Replace AU Paste AJ Justify AL Go To Line M-E Redo M-C Copy AB Where Was M-F Next

2. Vérifier les changements

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (master)
$ git log
commit e5473fe5aaa4c779bb9aff2f03361c4cbc04c100 (HEAD -> master)
Author: Ingbenchoud <benchoud17@gmail.com>
Date: Thu May 8 13:11:16 2025 -0400
    deuxieme modification
commit b2f95a35ee794623ba1e520c43e5c1e4c81eedb
Author: Ingbenchoud <benchoud17@gmail.com>
Date: Thu May 8 13:05:21 2025 -0400
    premiere modification
```

Exercice 3

1. Crée un fichier README.md

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (master)
$ touch README.md
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (master)
$ nano README.md
```

2. Ajoute et committe le fichier

```
fatal: pathspec 'README.' did not match any files
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~ (master)
$ git add README.md
fatal: pathspec 'README.' did not match any files
```


Exercice 4

1. Crée un nouveau repo sur GitHub sans README

2. Ajoute le remote et pousse ton projet

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 ~/mon-projet-git (main)
$ git remote add origin https://github.com/Ingbenchoud/mon-projet-git
```

Required fields are marked with an asterisk (*).

Owner *  Ingbenchoud / Repository name *

mon-projet-git is available.

Great repository names are short and memorable. Need inspiration? How about [solid-spork](#) ?

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☒ Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore
.gitignore template:


Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license
License:

Ingbenchoud / mon-projet-github

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

 mon-projet-github Public

Pin Unwatch 1 Fork 0 Star 0

main Branches Tags

+ <> Code

About

No description, website, or topics provided.

Readme

Activity

0 stars

1 watching

0 forks

Releases

No releases published
[Create a new release](#)

Packages

No packages published

Exercice 5

1. Crée une branche nouvelle-fonction

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 - (master)
$ git checkout -b nouvelle-fonction
Switched to a new branch 'nouvelle-fonction'
```

2. Ajoute un fichier fonction.py

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 - (nouvelle-fonction)
$ touch fonction.py
```

3. Merge cette branche dans main

```
Ing_Benchoud@BENCHOUD-2025 MINGW64 - (nouvelle-fonction)
$ git merge nouvelle-fonction
Already up to date.
```

Exercice 6

Gérer un conflit


```
Ing_Benchoud@BENCHOUD-2025 MINGW64 - (nouvelle-fonction)
$ git checkout -b index.txt
Switched to a new branch 'index.txt'

Ing_Benchoud@BENCHOUD-2025 MINGW64 - (index.txt)
$ nano index.txt

Ing_Benchoud@BENCHOUD-2025 MINGW64 - (index.txt)
$ git add index.txt
warning: in the working copy of 'index.txt', LF will be replaced by CRLF the next time Git touches it

Ing_Benchoud@BENCHOUD-2025 MINGW64 - (index.txt)
$ git commit -m "Ajout d'une ligne" -a
[index.txt 2c0a933] Ajout d'une ligne
1 file changed, 1 insertion(+)
create mode 100644 index.txt
```

Exercice7

 GIT FILE.txt

Add files via upload

now

 Refresh

 Activity