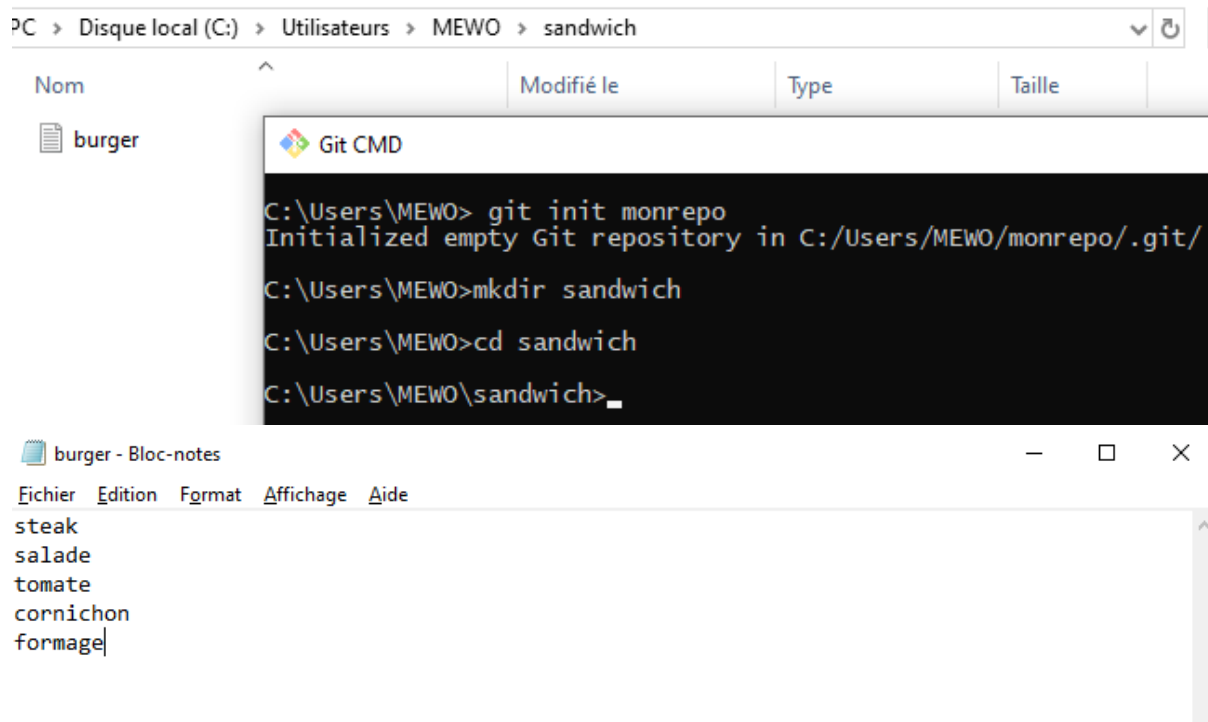


TP 4 : Git

2.1 Création d'un nouveau dépôt

Question 2.1 :

Je crée le répertoire sandwich, et y ajoute un bloc-notes (.txt) appelé **burger** qui contiendra la liste de ses ingrédients.



2.2 Add et Commit

Question 2.2 :

La console git nous informe qu'il existe un nouvel élément dans le fichier **sandwich** qui n'a pas encore été ajouté : **burger.txt**

```
C:\Users\MEWO\sandwich>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
   burger.txt

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

Question 2.3 :

Nous allons donc commencer par l'ajouter avec la commande **git add burger.txt**

```
C:\Users\MEWO\sandwich>git add burger.txt
```

La commande **git status** permet d'afficher l'état actuel du dépôt Git, en particulier les fichiers qui ont été modifiés, ajoutés ou supprimés par rapport au dernier commit.

```
C:\Users\MEWO\sandwich>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
   new file:   burger.txt
```

Puis, la commande **git diff --cached** permet de comparer l'index avec le dernier commit.

```
C:\Users\MEWO\sandwich>git diff --cached
diff --git a/burger.txt b/burger.txt
new file mode 100644
index 0000000..3875b72
--- /dev/null
+++ b/burger.txt
@@ -0,0 +1,5 @@
+steak
+salade
+tomate
+cornichon
+formage
\ No newline at end of file
```

Question 2.4 :

Nous allons ensuite **commit** (sauvegarder) les derniers changements sous le nom de "recette de burger"

```
C:\Users\MEWO\sandwich>git commit -m "<recette de burger>"
[master (root-commit) 76d08d7] <recette de burger>
1 file changed, 5 insertions(+)
create mode 100644 burger.txt
```

Question 2.5 :

Lorsque l'on utilise à nouveau la commande **git status**, il nous est indiqué qu'aucune modification n'a été apportée depuis le dernier **commit**, donc il n'a actuellement rien à **commit**.

```
C:\Users\MEWO\sandwich>git status
On branch master
nothing to commit, working tree clean
```

Question 2.6 :

Après l'utilisation de la commande **git log** nous pouvons observer :

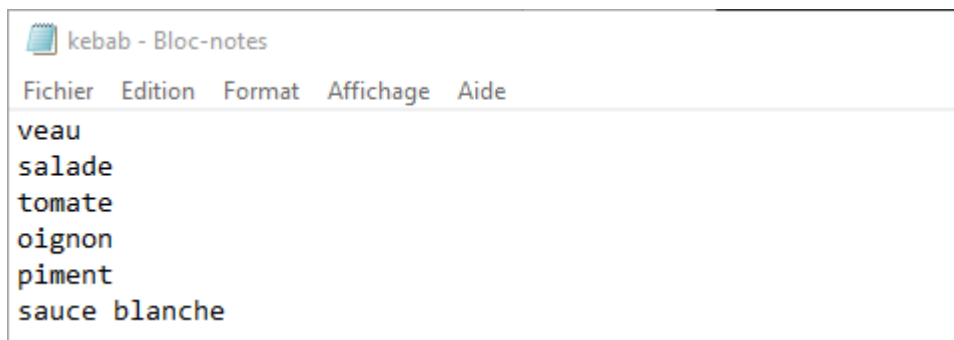
Une ligne, donc un changement effectué.

Et le numéro du **commit**, le 76d08d7eae1d50f1d1b794fe123f1da723f014ad.

```
C:\Users\MEWO\sandwich>git log
commit 76d08d7eae1d50f1d1b794fe123f1da723f014ad (HEAD -> master)
Author: unknown <theo.liss@mewo-campus.fr>
Date: Wed May 21 16:47:36 2025 +0200
```

Question 2.7 :

Nous effectuerons les mêmes étapes citées précédemment mais cette fois avec les fichiers textes **kebab.txt** et **tacos.txt**




```
C:\Users\MEWO\sandwich>git add kebab.txt
C:\Users\MEWO\sandwich>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   kebab.txt
```

```
C:\Users\MEWO\sandwich>git diff
```

```
C:\Users\MEWO\sandwich>git add .
```

```
C:\Users\MEWO\sandwich>git diff --cached
diff --git a/kebab.txt b/kebab.txt
new file mode 100644
index 0000000..42dcd4d
--- /dev/null
+++ b/kebab.txt
@@ -0,0 +1,6 @@
+veau
+salade
+tomate
+oignon
+piment
+sauce blanche
\ No newline at end of file
```

```
C:\Users\MEWO\sandwich>git commit -m "<recette de kebab>"
[master c949014] <recette de kebab>
1 file changed, 6 insertions(+)
create mode 100644 kebab.txt
```

 tacos - Bloc-notes

Fichier Edition Format Affichage Aide

```
tenders
viande hachée de boeuf
frites
sauce fromagère
```

```
C:\Users\MEWO\sandwich>git add tacos.txt
C:\Users\MEWO\sandwich>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   tacos.txt

C:\Users\MEWO\sandwich>git diff
C:\Users\MEWO\sandwich>git add .
C:\Users\MEWO\sandwich>git diff --cached
diff --git a/tacos.txt b/tacos.txt
new file mode 100644
index 0000000..edbe887
--- /dev/null
+++ b/tacos.txt
@@ -0,0 +1,4 @@
+tenders
+viande hachée de boeuf
+frites
+sauce fromagère
\ No newline at end of file
C:\Users\MEWO\sandwich>git commit -m "<recette de tacos>"
[master f553d8e] <recette de tacos>
1 file changed, 4 insertions(+)
```

Question 2.8 :

Avec la commande **git log** nous pouvons observer nos trois fichiers textes qui ont été commit avec chacun le nom de la recette attribuée.

```
C:\Users\MEWO\sandwich>git log
commit f553d8ef5625adfa1e4bc6bfa5c30551e621371d (HEAD -> master)
Author: unknown <theo.liss@mewo-campus.fr>
Date: Sun Jun 15 16:12:56 2025 +0200

    <recette de tacos>

commit c9490148be18f58dc788d8f6bc886f890742b2e2
Author: unknown <theo.liss@mewo-campus.fr>
Date: Sun Jun 15 16:07:58 2025 +0200

    <recette de kebab>

commit 76d08d7eae1d50f1d1b794fe123f1da723f014ad
Author: unknown <theo.liss@mewo-campus.fr>
Date: Wed May 21 16:47:36 2025 +0200

    <recette de burger>
```

Encore une fois **git status** nous indique qu'il n'y a rien à **commit**.

```
C:\Users\MEWO\sandwich>git status
On branch master
nothing to commit, working tree clean
```

La commande **git log --graph --pretty=short** permet d'afficher l'historique des **commits** avec la structure des branches comme une arborescence.

```
C:\Users\MEWO\sandwich>git log --graph --pretty=short
* commit f553d8ef5625adfa1e4bc6bfa5c30551e621371d (HEAD -> master)
  Author: unknown <theo.liss@mewo-campus.fr>

     <recette de tacos>
* commit c9490148be18f58dc788d8f6bc886f890742b2e2
  Author: unknown <theo.liss@mewo-campus.fr>

     <recette de kebab>
* commit 76d08d7eae1d50f1d1b794fe123f1da723f014ad
  Author: unknown <theo.liss@mewo-campus.fr>

     <recette de burger>
```

Ensuite, pour les commandes **gitg** et **gitk** elles permettent d'afficher une interface graphique pour explorer l'historique de Git.

Contrairement à **gitk**, **gitg** possède une interface plus moderne et est intégrée à GNOME (sous Linux).

```
C:\Users\MEWO\sandwich>gitk
```

🔴🔴🔴 sandwich: All files - gitk

File Edit View Help

🟡 **master** <recette de tacos> unknown <theo.liss@mewo-campus.f 2025-06-15 16:12:56
 🔵 <recette de kebab> unknown <theo.liss@mewo-campus.f 2025-06-15 16:07:58
 🔵 <recette de burger> unknown <theo.liss@mewo-campus.f 2025-05-21 16:47:36

Commit ID: f553d8ef5625adfale4bc6bfa5c30551e621371d Row 1 / 3

Find commit containing: Search Patch Tree

Diff Old version New version Lines of context: 3 Ignore space changes **Comments**

Author: unknown <theo.liss@mewo-campus.fr> 2025-06-15 16:12:56
 Committer: unknown <theo.liss@mewo-campus.fr> 2025-06-15 16:12:56
 Parent: c9490148be18f58dc788d8f6bc886f890742b2e2 (<recette de kebab>)
 Branch: master
 Follows:
 Precedes:

<recette de tacos>

----- tacos.txt -----
 new file mode 100644
 index 0000000..edbe887
 @@ -0,0 +1,4 @@
 +tenders
 +viande hachée de boeuf
 +frites
 +sauce fromagère
 \ No newline at end of file

2.3 Voyage dans le temps

Question 2.9 /10/11

La commande **git reset** permet de revenir en arrière dans l'historique Git en modifiant l'index et en annulant le dernier **commit**.

```
C:\Users\MEWO\sandwich>git reset burger.txt
C:\Users\MEWO\sandwich>git status
On branch master
nothing to commit, working tree clean

C:\Users\MEWO\sandwich>git checkout c9490148be18f58dc788d8f6bc886f890742b2e2
Note: switching to 'c9490148be18f58dc788d8f6bc886f890742b2e2'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

  git switch -c <new-branch-name>

Or undo this operation with:

  git switch -

Turn off this advice by setting config variable advice.detachedHead to false
HEAD is now at c949014 <recette de kebab>

C:\Users\MEWO\sandwich>git status
HEAD detached at c949014
nothing to commit, working tree clean
```

Question 2.12 :

Git checkout master nous aide à définir une branche principale.

```
C:\Users\MEW0\sandwich>git checkout master
Previous HEAD position was c949014 <recette de kebab>
Switched to branch 'master'

C:\Users\MEW0\sandwich>git status
On branch master
nothing to commit, working tree clean
```