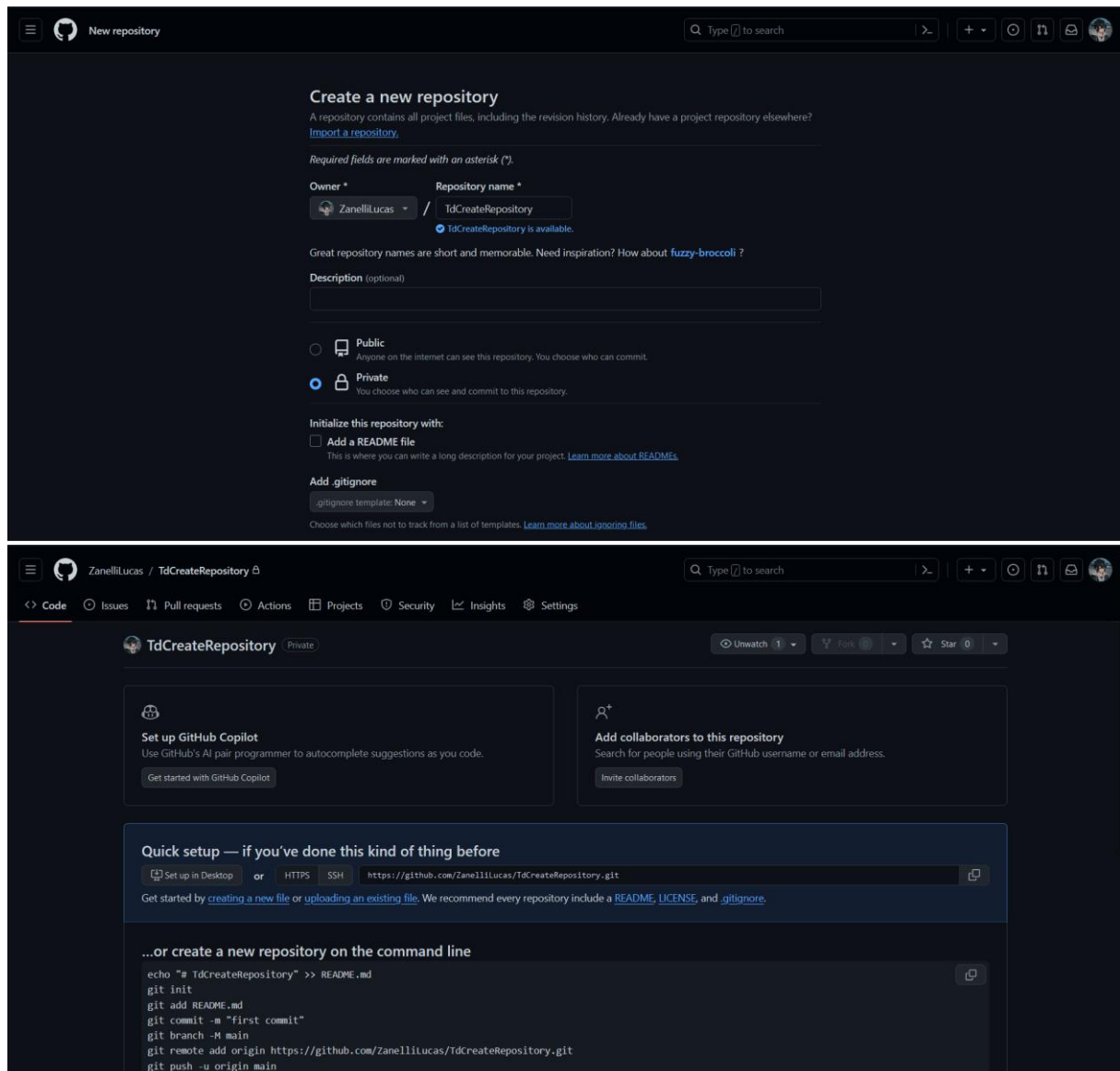


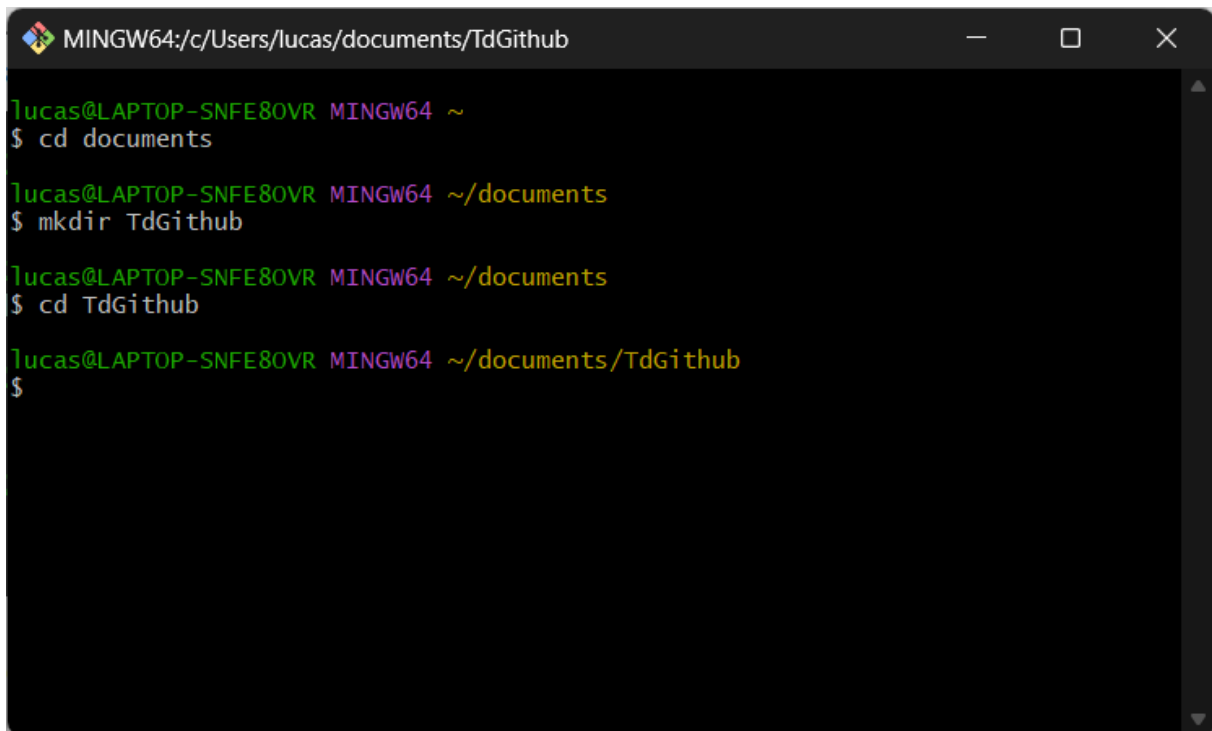
Compte rendu GitHub

Zanelli Lucas

Pour Commencer je me suis créé un compte sur GitHub, ensuite j'ai créé une nouvelle repository et en le mettant en privée.



Ensuite j'ai lancé Git Bash et je me suis positionné dans les documents, puis j'ai créé un répertoire que j'ai nommé TdGithub et je suis rentré dans le répertoire en tapant la commande `cd TdGithub`.



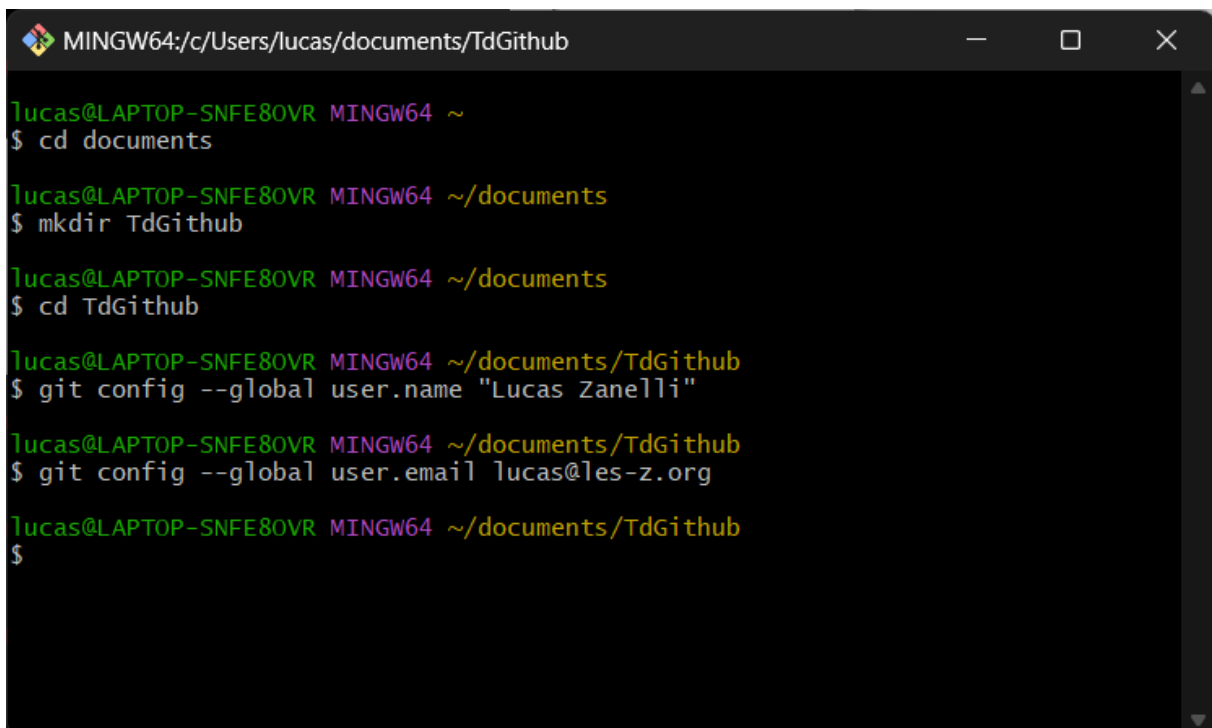
```
MINGW64:/c/Users/lucas/documents/TdGithub
Lucas@LAPTOP-SNFE80VR MINGW64 ~
$ cd documents

Lucas@LAPTOP-SNFE80VR MINGW64 ~/documents
$ mkdir TdGithub

Lucas@LAPTOP-SNFE80VR MINGW64 ~/documents
$ cd TdGithub

Lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$
```

Par la suite j'ai configuré mon identité, et ensuite mon adresse email.



```
MINGW64:/c/Users/lucas/documents/TdGithub
Lucas@LAPTOP-SNFE80VR MINGW64 ~
$ cd documents

Lucas@LAPTOP-SNFE80VR MINGW64 ~/documents
$ mkdir TdGithub

Lucas@LAPTOP-SNFE80VR MINGW64 ~/documents
$ cd TdGithub

Lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global user.name "Lucas Zanelli"

Lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global user.email lucas@les-z.org

Lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$
```

J'ai configuré les couleurs.

```
MINGW64:/c/Users/lucas/documents/TdGithub
lucas@LAPTOP-SNFE80VR MINGW64 ~/documents
$ mkdir TdGithub

lucas@LAPTOP-SNFE80VR MINGW64 ~/documents
$ cd TdGithub

lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global user.name "Lucas Zanelli"

lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global user.email lucas@les-z.org

lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global color.diff auto

lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global color.status auto

lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global color.branch auto

lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$
```

Ensuite j'ai configuré les éditeurs

```
lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global core.editor vim

lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub
$ git config --global merge.tool vimdiff
```

Puis j'ai initialisé mon projet git

```
lucas@LAPTOP-SNFE80VR MINGW64 ~/documents/TdGithub (NewCss)
$ git init
Reinitialized existing Git repository in C:/Users/lucas/Documents/TdGithub/.git/
```

Pour commencer j'ai taper la commande pwd et ensuite je suis aller dans le dossier TdGithub

```
MINGW64:/c/Users/lucas/Documents/TdGithub
Lucas@LAPTOP-SNFE80VR MINGW64 ~
$ pwd
/c/Users/lucas

Lucas@LAPTOP-SNFE80VR MINGW64 ~
$ cd Documents

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents
$ cd TdGithub

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$
```

Ensuite j'ai ajouté un fichier index.html et un fichier styles.css dans le dossier TdGithub, puis après j'ai vérifié qu'ils soient bien présents dans mon dossier TdGithub

```
MINGW64:/c/Users/lucas/Documents/TdGithub
$ cd TdGithub

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ touch index.html

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ ls
index.html  styles.css

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html
    styles.css

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

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$
```

```
MINGW64:/c/Users/lucas/Documents/TdGithub
new file:   index.html

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

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ git add index.html styles.css

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   index.html
        new file:   styles.css

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$
```

Ensuite j'ai refait un git commit

```
lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ git commit -m "Ajout et remplissage des fichiers index.html et styles.css"
[master (root-commit) 3a53add] Ajout et remplissage des fichiers index.html et styles.css
2 files changed, 13 insertions(+)
create mode 100644 index.html
create mode 100644 styles.css

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ |
```

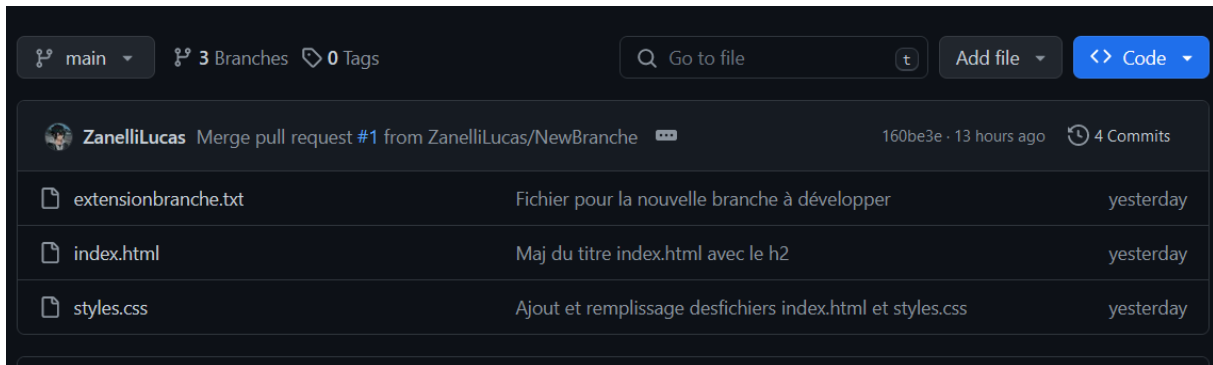
Puis j'ai tapé les commandes suivantes pour stocker le projet dans le cloud

```
lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ git remote add origin https://github.com/ZanelliLucas/TdCreateRepository

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (master)
$ git branch -M main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ |
```

Voici une partie de l'évolution.



Après je crée une nouvelle branche et je switch dedans pour créer un nouveau fichier

```
MINGW64:/c/Users/lucas/Documents/TdGithub
git: 'branche' is not a git command. See 'git --help'.

The most similar command is
  branch

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ git branch
  NewBranche
* main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ git checkout NewBranche
Switched to branch 'NewBranche'

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewBranche)
$ git branch
* NewBranche
  main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewBranche)
$ touch extensionbranche.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewBranche)
$ |
```

Ensuite j'ai indexé le fichier et fait un commit de se même fichier, puis je l'ai envoyez sur GitHub

```
MINGW64:/c/Users/lucas/Documents/TdGithub
$ git commit -m "Fichier pour la nouvelle branche à développer"
[NewBranche 9c40d24] Fichier pour la nouvelle branche à développer
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 extensionbranche.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewBranche)
$ git push -u origin NewBranche
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 356 bytes | 356.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'NewBranche' on GitHub by visiting:
remote:   https://github.com/ZanelliLucas/TdCreateRepository/pull/new/NewBranche
remote:
To https://github.com/ZanelliLucas/TdCreateRepository
 * [new branch]      NewBranche -> NewBranche
branch 'NewBranche' set up to track 'origin/NewBranche'.

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewBranche)
$ |
```

Ensuite je vais fusionner la branche main avec NewBranche

```
MINGW64:/c/Users/lucas/Documents/TdGithub
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 356 bytes | 356.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'NewBranche' on GitHub by visiting:
remote:   https://github.com/ZanelliLucas/TdCreateRepository/pull/new/NewBranche
remote:
To https://github.com/ZanelliLucas/TdCreateRepository
 * [new branch]      NewBranche -> NewBranche
branch 'NewBranche' set up to track 'origin/NewBranche'.

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewBranche)
$ git checkout main
Switched to branch 'main'

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ git branch
NewBranche
* main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ |
```

J'ai effectuer un merge des branche

```
Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ git merge NewBranche
Updating 228f924..9c40d24
Fast-forward
 extensionbranche.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 extensionbranche.txt

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$
```

Pour récupérer des codes je dois utiliser une commande de clone, et ensuite je dois récupérer des modifications sur la branche main

```
Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ git clone https://github.com/ZanelliLucas/TdCreateRepository.git
Cloning into 'TdCreateRepository'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 10 (delta 2), reused 9 (delta 1), pack-reused 0
Receiving objects: 100% (10/10), done.
Resolving deltas: 100% (2/2), done.

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ git pull origin main
From https://github.com/ZanelliLucas/TdCreateRepository
 * branch      main      -> FETCH_HEAD
Already up to date.

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (main)
$ |
```

Après cela j'ai dû créer une nouvelle branche nommée NewCss

```
Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewCss)
$ git add styles2.css

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewCss)
$ git push -u origin NewCss
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'NewCss' on GitHub by visiting:
remote:   https://github.com/ZanelliLucas/TdCreateRepository/pull/new/NewCss
remote:
To https://github.com/ZanelliLucas/TdCreateRepository
 * [new branch]      NewCss -> NewCss
branch 'NewCss' set up to track 'origin/NewCss'.

Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewCss)
$ |
```


Ensuite j'ai indexé le fichier index.html et je l'ai commit

```
MINGW64:/c/Users/lucas/Documents/TdGithub
Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewCss)
$ git push -u origin NewCss
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'NewCss' on GitHub by visiting:
remote:   https://github.com/ZanelliLucas/TdCreateRepository/pull/new/NewCss
remote:
To https://github.com/ZanelliLucas/TdCreateRepository
 * [new branch]      NewCss -> NewCss
branch 'NewCss' set up to track 'origin/NewCss'.


Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewCss)
$ git add index.html


Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewCss)
$ git commit -m "changement de styles.css par styles2.css dans le code de index.html"
[NewCss f10651e] changement de styles.css par styles2.css dans le code de index.html
2 files changed, 1 insertion(+), 1 deletion(-)
create mode 100644 styles2.css


Lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/TdGithub (NewCss)
$ |
```


Et j'ai donc pu créer une pull request

Add more commits by pushing to the `NewBranche` branch on `ZanelliLucas/TdCreateRepository`.



**Require approval from specific reviewers before merging**
[Rulesets](#) ensure specific people approve pull requests before they're merged.

**Continuous integration has not been set up**
[GitHub Actions](#) and [several other apps](#) can be used to automatically catch bugs and enforce style.

**This branch has no conflicts with the base branch**
Merging can be performed automatically.

Merge pull request


You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Conversation 0

Commits 1


Checks 0


Files changed 1





ZanelliLucas commented now


change h1 color from red to blueviolet






 Fichier pour la nouvelle branche à développer 9c40d24



 ZanelliLucas merged commit `160be3e` into `main` now

Revert




Pull request successfully merged and closed
You're all set—the `NewBranche` branch can be safely deleted.

Delete branch

Code

Blame

11 lines (11 loc) · 248 Bytes

 Code 55% faster with GitHub Copilot

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title></title>
5   <link rel="stylesheet" type="text/css" href="styles2.css">
6 </head>
7 <body>
8   <h1>Apprendissage de manipulation de Git et GitHub</h1>
9   <h2>Gestions des versions</h2>
10 </body>
11 </html>
```

Ensuite j'ai dû créer branche principale

```
MINGW64:/c/Users/lucas/Documents/Test
```

```
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the commit.  
#  
# On branch master  
#  
# Initial commit  
#  
# Changes to be committed:  
#   new file:   PremierFichier.txt  
#  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
  
.git/COMMIT_EDITMSG [unix] (10:03 26/02/2024) 1,0-1 All  
~/Documents/Test/.git/COMMIT_EDITMSG" [unix] 11L, 240B
```

J'ai ensuite créé une branche `branchTest`, après cela je me suis rendu compte que j'avais master alors que son doit être main donc j'ai modifié cela

```
MINGW64:/c:/Users/lucas/Documents/test1

[master (root-commit) ca3e93d] création de la branche master
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 PremierFichier.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (master)
$ git branch brancheTest

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (master)
$ git branch
  brancheTest
* master

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (master)
$ git branch -M main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git branch
  brancheTest
* main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ |
```

J'ai ensuite supprimé la branche brancheTest, puis après j'ai vérifié le status de ma branche en modifiant cela

```
MINGW64:/c/Users/lucas/Documents/test1
$ git branch
  brancheTest
* main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git branch -d brancheTest
Deleted branch brancheTest (was ca3e93d).

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git branch
* main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git status
On branch main
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:   PremierFichier.txt

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

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ |
```

```
MINGW64:/c/Users/lucas/Documents/test1

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

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git checkout brancheTest
error: pathspec 'brancheTest' did not match any file(s) known to git

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git branch -D brancheTest
error: branch 'brancheTest' not found

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git add PremierFichier.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   PremierFichier.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ |
```

Après j'ai créé une remise pour mettre de côté les modifications

```
MINGW64:/c/Users/lucas/Documents/test1
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   PremierFichier.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git stash
Saved working directory and index state WIP on main: ca3e93d création de la bran
che master

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git status
On branch main
nothing to commit, working tree clean

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git branch brancheCommit

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git checkout brancheCommit
Switched to branch 'brancheCommit'

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (brancheCommit)
$
```

Puis j'ai créé une nouvelle branche

```
MINGW64:/c/Users/lucas/Documents/test1
        modified:   PremierFichier.txt

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

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (brancheCommit)
$ git log
commit ca3e93da56924bfde4ff3271b8005dbc31541b5a (HEAD -> brancheCommit, main)
Author: Lucas Zanelli <lucas@les-z.org>
Date:   Mon Feb 26 11:12:37 2024 +0100

    création de la branche master

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (brancheCommit)
$ git branch
* brancheCommit
  main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (brancheCommit)
$ git checkout main
Switched to branch 'main'
M       PremierFichier.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$
```

Ensuite je me suis mis sur la branche principale, pour recréer une nouvelle branche que j'ai nommé newbranche.

```
MINGW64:/c/Users/lucas/Documents/test1

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (brancheCommit)
$ git branch
* brancheCommit
  main

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (brancheCommit)
$ git checkout main
Switched to branch 'main'
M       PremierFichier.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git reset --hard HEAD
HEAD is now at ca3e93d création de la branche master

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git branch newbranche

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git checkout newbranche
Switched to branch 'newbranche'

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$
```

Ensuite je me suis mis sur la nouvelle branche newbranche pour faire un git reset

```
MINGW64:/c/Users/lucas/Documents/test1

$ git branch newbranche

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (main)
$ git checkout newbranche
Switched to branch 'newbranche'

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git reset --hard ca3e93d
HEAD is now at ca3e93d création de la branche master

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ touch fic..txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git status
On branch newbranche
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    fic..txt

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

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$
```

Après j'ai créé un nouveau fichier et j'ai ajouter

```
MINGW64:/c/Users/lucas/Documents/test1
$ touch fic..txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git status
On branch newbranche
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        fic..txt

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

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git add fic..txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git status
On branch newbranche
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   fic..txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ |
```

Puis j'ai fait un git commit

```
MINGW64:/c/Users/lucas/Documents/test1

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git status
On branch newbranche
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   fic..txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git commit -m "commit 1"
[newbranche 087fb50] commit 1
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fic..txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git commit --amend -m "rajout du fichier fic..txt sur newbranche"
[newbranche 55c159c] rajout du fichier fic..txt sur newbranche
Date: Mon Feb 26 11:32:08 2024 +0100
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fic..txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ |
```

Après cela j'ai fait un git log

```
MINGW64:/c/Users/lucas/Documents/test1

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git commit --amend -m "rajout du fichier fic..txt sur newbranche"
[newbranche 55c159c] rajout du fichier fic..txt sur newbranche
Date: Mon Feb 26 11:32:08 2024 +0100
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fic..txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git log
commit 55c159cea2f96da0e13760e8d67cd7d769f4bc77 (HEAD -> newbranche)
Author: Lucas Zanelli <lucas@les-z.org>
Date: Mon Feb 26 11:32:08 2024 +0100

    rajout du fichier fic..txt sur newbranche

commit ca3e93da56924bfde4ff3271b8005dbc31541b5a (main, brancheCommit)
Author: Lucas Zanelli <lucas@les-z.org>
Date: Mon Feb 26 11:12:37 2024 +0100

    création de la branche master

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ |
```

Ensuite j'ai créé un fichier et je l'ai ajouté dans ma branche newbranche

```
MINGW64:/c/Users/lucas/Documents/test1

    rajout du fichier fic..txt sur newbranche

commit ca3e93da56924bfde4ff3271b8005dbc31541b5a (main, brancheCommit)
Author: Lucas Zanelli <lucas@les-z.org>
Date: Mon Feb 26 11:12:37 2024 +0100

    création de la branche master

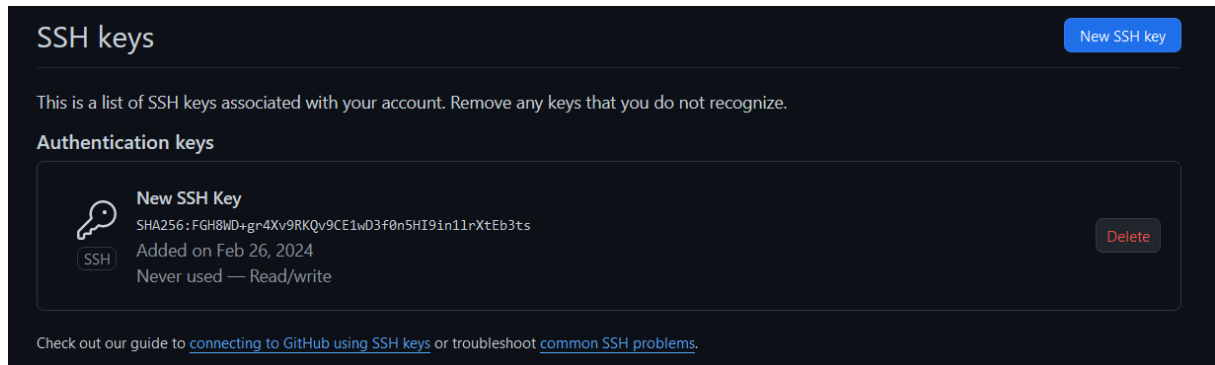
lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ touch ficoublie.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git add ficoublie.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ git commit --amend --no-edit
[newbranche e4a1836] rajout du fichier fic..txt sur newbranche
Date: Mon Feb 26 11:32:08 2024 +0100
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fic..txt
create mode 100644 ficoublie.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/test1 (newbranche)
$ |
```


J'ai créé une nouvelle clé



Après avoir ajouté le fichier dans la branche j'ai regardé les modifications que j'ai faite au fichier

```
MINGW64:/c/Users/lucas/Documents/Test1
newbranche
lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/Test1 (main)
$ git add PremierFichier.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/Test1 (main)
$ git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   PremierFichier.txt

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/Test1 (main)
$ git commit -m "changement contenu"
[main 4ccc89f] changement contenu
1 file changed, 1 insertion(+)

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/Test1 (main)
$ git blame PremierFichier.txt
4ccc89f8 (Lucas Zane1li 2024-02-26 14:56:19 +0100 1) ddddd

lucas@LAPTOP-SNFE80VR MINGW64 ~/Documents/Test1 (main)
$ |
```