

Rapport brief project Git

scenario#1

- First step: Immersion

1- Création de repo “/projets”:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop (master)
$ mkdir projects
```

2- Vérification du status:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

3- Creation de fichier README.md et l'ajout de la ligne ‘#Demo project un simple fichier’:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ touch README.md

youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ vim README.md
```



4- Le staging et le committing:

```

youcode@DESKTOP-MJF171K MINGW64 ~/Desktop/projects/demo (master)
$ git add .

youcode@DESKTOP-MJF171K MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "README MODIF"
[master (root-commit) d4eb55c] README MODIF
1 file changed, 1 insertion(+)
create mode 100644 README.md

youcode@DESKTOP-MJF171K MINGW64 ~/Desktop/projects/demo (master)
$

```

- Second step: La découverte

1- Déplacement dans les fichiers de configuration “.git”:

```

Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ cd .git

Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo/.git (GIT_DIR!)
$

```

2- Ls --al:

```

youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo/.git (GIT_DIR!)
$ ls -al
total 17
drwxr-xr-x 1 youcode 197121  0 nov. 26 11:47 ./
drwxr-xr-x 1 youcode 197121  0 nov. 26 11:47 ../
-rw-r--r-- 1 youcode 197121 13 nov. 26 11:47 COMMIT_EDITMSG
-rw-r--r-- 1 youcode 197121 73 nov. 26 11:46 config
-rw-r--r-- 1 youcode 197121 23 nov. 26 11:46 description
-rw-r--r-- 1 youcode 197121  0 nov. 26 11:46 HEAD
drwxr-xr-x 1 youcode 197121  0 nov. 26 11:46 hooks/
-rw-r--r-- 1 youcode 197121 137 nov. 26 11:47 index
drwxr-xr-x 1 youcode 197121  0 nov. 26 11:46 info/
drwxr-xr-x 1 youcode 197121  0 nov. 26 11:47 logs/
drwxr-xr-x 1 youcode 197121  0 nov. 26 11:47 objects/
drwxr-xr-x 1 youcode 197121  0 nov. 26 11:46 refs/

```

3- Le retour vers le repo et la création du fichier “Licence.md”:

```

youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo/.git (GIT_DIR!)
$ cd ..

youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ touch Licence.md

```

4- Le commit:

```

youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git add .

youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "add licence"
[master 928f7a1] add licence
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Licence.md

```

5- L’affiche des fichiers traqués:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git ls-tree -r master --name-only
Licence.md
README.md
```

- Third step: Historique

1- L'affiche de dernier commit:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git log -1 --oneline
c8aa1be (HEAD -> master) add licence.md
```

2- La création d'alias:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git config --global alias.historique 'log -1 --oneline'

Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git historique
c8aa1be (HEAD -> master) add licence.md
```

3- L'affiche de la liste d'alias:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git config --get-regexp alias
alias.historique log -1 --oneline
```

4- L'affichage de l'historique des commit du fichier README.md avec l'alias:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git historique 40bd39
40bd396 README modif
```

- Fourth step: Excluding files

1- Renommer le fichier Licence.md à Licence.txt:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ mv Licence.md Licence.txt
```

2- Le staging et le mise à jour:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git add Licence.txt
```

3- Le commit:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m 'change name licence'
[master 99c830c] change name licence
1 file changed, 0 insertions(+), 0 deletions(-)
rename Licence.md => Licence.txt (100%)
```

4- La création d'un fichier application.log:

```
Youcode@DESKTOP-MJF171K MINGW64 ~/Desktop/projects/demo (master)
$ touch application.log
```

5- L'ajout de la ligne "*.log":



```
MINGW64: c:/Users/youcode/Desktop/projects/demo
*.log

.gitignore [unix] (14:27 26/11/2019) 1,5 Tout
".gitignore" [unix] 1L, 6C
```

6- Le staging et le commit:

```
Youcode@DESKTOP-MJF171K MINGW64 ~/Desktop/projects/demo (master)
$ git add .

Youcode@DESKTOP-MJF171K MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m 'add gitignore & modif'
[master 5cc87a2] add gitignore & modif
2 files changed, 1 insertion(+)
create mode 100644 .gitignore
delete mode 100644 Licence.md
```

- Fifth step: Branching and Merging

1- La modification du fichier README.md:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ vim README.md
```


2- Le staging et le commit:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (updates)
$ git commit -am 'change readme'
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory
[updates 8105365] change readme
1 file changed, 1 insertion(+), 1 deletion(-)
```

3- L'affichage d'historique d'alias:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (updates)
$ git historique
* 8105365 (HEAD -> updates) change readme
* d90174b (master) create gitignore
* fd29e97 renommer licence
* b41878b create licence
* 6ed2ee1 create readme
```

4- Le retour ver la branche master:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (updates)
$ git checkout master
Switched to branch 'master'
```

5- L'affichage de l'historique avec l'alias:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git historique
* 8105365 (updates) change readme
* d90174b (HEAD -> master) create gitignore
* fd29e97 renommer licence
* b41878b create licence
* 6ed2ee1 create readme
```

6- Le merge:

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git merge updates
Updating d90174b..8105365
Fast-forward
 README.md | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)
```

- Sixth step: Conflict Resolution

1- La création de la branche "BAD":

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git branch BAD
```

2- La modification du fichier README.md et l'ajout de la ligne "trouble":

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ vim README.md
```

3- Le staging et le commit:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git checkout BAD
Switched to branch 'BAD'
M       README.md

Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (BAD)
$ git commit -am 'modif readme 2'
[BAD 4a59673] modif readme 2
1 file changed, 2 insertions(+), 1 deletion(-)
```

4- La correction de commentaire du dernier commit:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git commit --amend
[master f630fe7] Modif README 3
Date: Tue Nov 26 16:07:57 2019 +0100
1 file changed, 2 insertions(+), 1 deletion(-)
```

5- L'affichage d'historique:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git historique
f630fe7 (HEAD -> master) Modif README 3
```

6- Le merge de la branche "BAD":

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master)
$ git merge BAD
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
```

7- L'exécution de la commande "cat README.md":

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ cat README.md
#Demo project un simple fichier modifié
<<<<<<< HEAD
Troubleshooting
=====
Trouble
>>>>>>> BAD
```

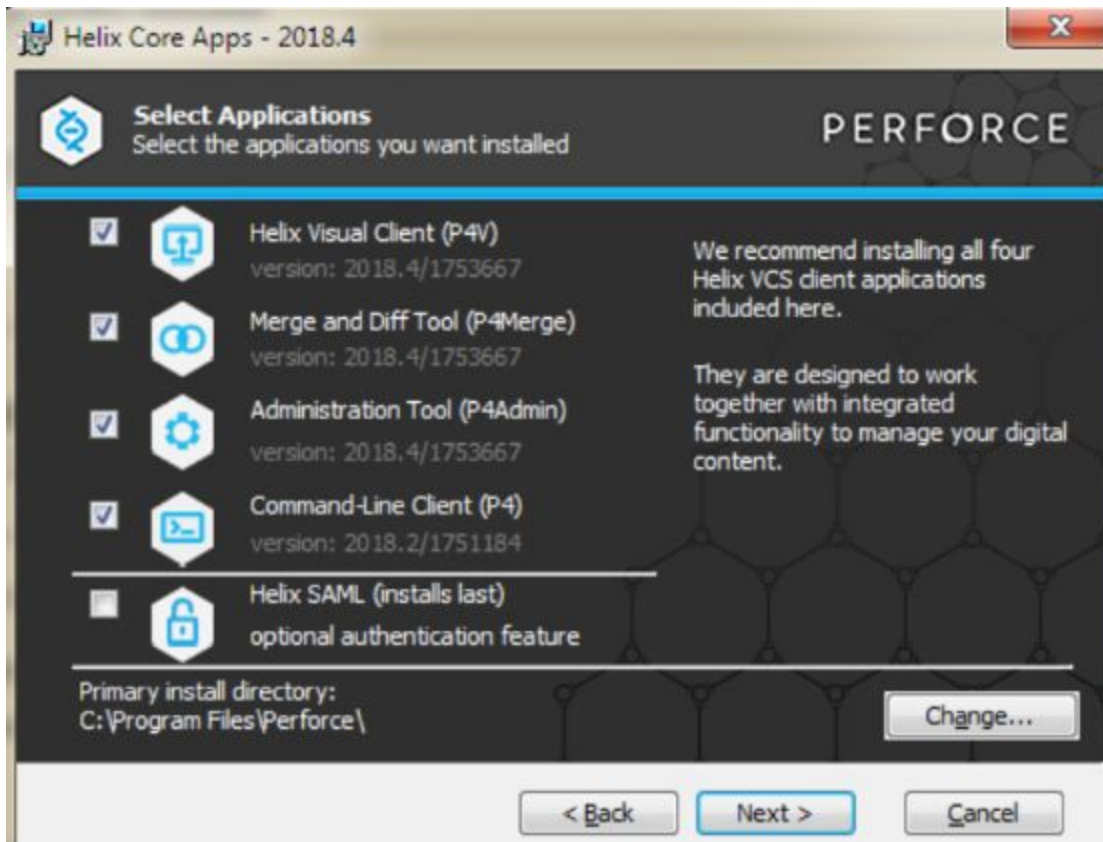
8- la commande "git mergetool" :

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git mergetool
Merging:
README.md

Normal merge conflict for 'README.md':
{local}: modified file
{remote}: modified file
The merge tool p4merge is not available as 'C:\Program Files\Perforce\p4merge.exe'
```

- Seventh step: merge tools

1- L'installation de p4merge:



2- La commande Git config --global merge.tool p4merge :

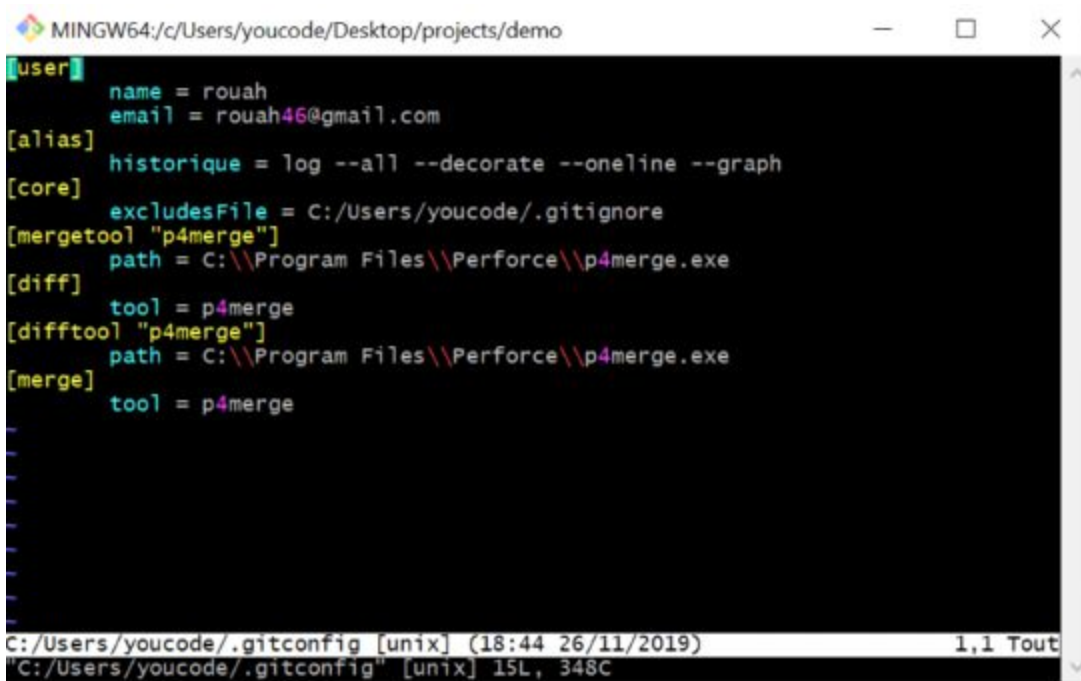
```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git config --global merge.tool p4merge
```

3- Git config --global mergetool.p4merge.path "lien d'installation" (.exe):

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git config --global mergetool.p4merge.path "C:\Program Files\Perforce\p4merge.exe"
```

4- La configuration du prompt:


```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git config --global --edit
```

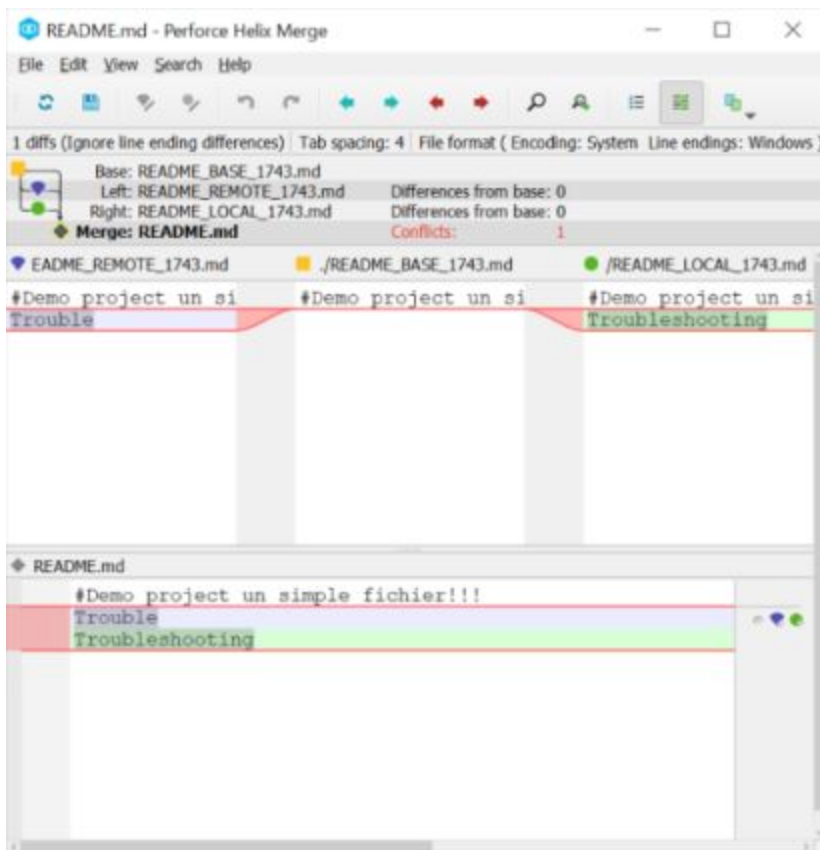


The screenshot shows a Windows command prompt window titled "MINGW64: c:/Users/youcode/Desktop/projects/demo". The prompt is at the user's shell, and the user has entered the command `git config --global --edit`. The output is a text editor displaying the contents of the global git configuration file, `C:/Users/youcode/.gitconfig`. The configuration includes user information, aliases, core settings, and merge/diff tools. At the bottom, a status bar indicates the file is at line 1, column 1.

```
[user]
  name = rouah
  email = rouah46@gmail.com
[alias]
  historique = log --all --decorate --oneline --graph
[core]
  excludesFile = C:/Users/youcode/.gitignore
[mergetool "p4merge"]
  path = C:\\Program Files\\Perforce\\p4merge.exe
[diff]
  tool = p4merge
[difftool "p4merge"]
  path = C:\\Program Files\\Perforce\\p4merge.exe
[merge]
  tool = p4merge

C:/Users/youcode/.gitconfig [unix] (18:44 26/11/2019) 1,1 Tout
"C:/Users/youcode/.gitconfig" [unix] 15L, 348C
```

5- la commande git mergetool:



- **Eighth step:** Challenge

1- L'écriture de la clause pour rejeter les fichiers indésirables et redondants:

```
Youcode@DESKTOP-TLL2MM MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ vim .gitignore
```

[illegible]

scenario#2

- **First step**: Tagging

1- Le déplacement sur la branche principale et la création de Tag V1.0 :

```
youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git tag -a v1.0 -m 'RELEASE 1.0'
```

2- L'affichage des informations sur le Tag:

```

youcode@DESKTOP-KVTCEUL MINGW64 ~/Desktop/projects/demo (master)
$ git show v1.0
tag v1.0
Tagger: rouah <rouah46@gmail.com>
Date:   Wed Nov 27 10:59:12 2019 +0100

RELEASE 1.0

commit 37de517c88345990c307a9e87f176a94b4257236 (HEAD -> master, tag: v1.0)
Merge: 13e1bff 4009b2f
Author: rouah <rouah46@gmail.com>
Date:   Wed Nov 27 09:44:59 2019 +0100

    4

diff --cc .gitignore
index 397b4a7,397b4a7..3598a4d
--- a/.gitignore
+++ b/.gitignore
@@ -1,1 -1,1 +1,2 @@
*.log
+*.orig

```

- Seventh step: Fetch and pull

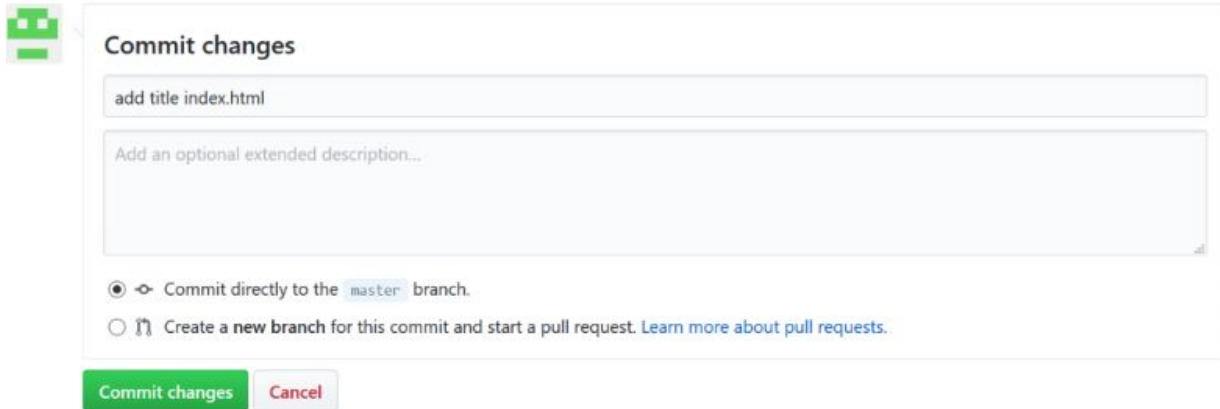
1- L'édit du fichier Index.html sur github:

```

91 lines (83 sloc) 4.13 KB
Raw Blame History
1 <!doctype html>
2 <html class="no-js" lang="">
3 <head>
4 <meta charset="utf-8">
5 <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
6 <title>Mon premier sit web</title>
7 <meta name="description" content="">
8 <meta name="viewport" content="width=device-width, initial-scale=1">
9
10 <link rel="stylesheet" href="css/bootstrap.min.css">
11 <style>
12     body {
13         padding-top: 50px;
14         padding-bottom: 20px;
15     }
16 </style>
17 <link rel="stylesheet" href="css/bootstrap-theme.min.css">
18 <link rel="stylesheet" href="css/main.css">
19
20 <!--[if lt IE 9]>
21     <script src="js/vendor/html5-3.6-respons-1.4.2.min.js"></script>
22 <![endif]-->
23 </head>
24 <body>
25 <nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">
26 <div class="container">

```

2- Le commit sur github:



Commit changes

add title index.html

Add an optional extended description...

☒ Commit directly to the `master` branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes **Cancel**

3- L'édit du fichier README.md sur le git et le repo local et le staging et le commit:

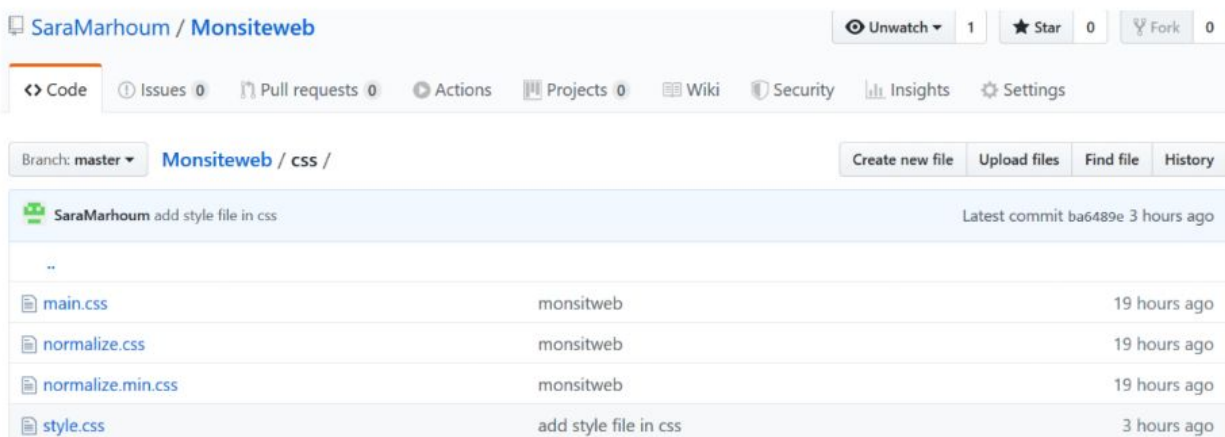
```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/Monsiteweb-local (master)
$ vim README.md

Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/Monsiteweb-local (master)
$ git commit -am 'edit README.md'
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory
[master b7a0c47] edit README.md
1 file changed, 2 insertions(+), 1 deletion(-)
```

scenario#3

- First step: Changes on Github

1- L'ajout du fichier style.css sur le dossier CSS.



SaraMarhoum / Monsiteweb

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Branch: master Monsiteweb / css / Create new file Upload files Find file History

SaraMarhoum add style file in css Latest commit ba6489e 3 hours ago

File	Commit	Time
main.css	monsitweb	19 hours ago
normalize.css	monsitweb	19 hours ago
normalize.min.css	monsitweb	19 hours ago
style.css	add style file in css	3 hours ago

2- L'ajout de la balise <H1> : <h1> sur Index.html :


```
25 lines (21 sloc) | 891 Bytes
Raw Blame History

1 <!doctype html>
2 <html class="no-js" lang="">
3   <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
6     <title>mon premier site web </title>
7     <meta name="description" content="">
8     <meta name="viewport" content="width=device-width, initial-scale=1">
9
10    <link rel="stylesheet" href="css/normalize.min.css">
11    <link rel="stylesheet" href="css/main.css">
12
13    <!--[if lt IE 9]>
14      <script src="https://cdnjs.cloudflare.com/ajax/libs/html5shiv/3.7.3/html5shiv.min.js"></script>
15      <script>window.html5 || document.write('<script src="js/vendor/html5shiv.js"></script>')</script>
16    <![endif]-->
17  </head>
18  <body>
19    <h1> modification récente </h1>
20
21    <p>Hello world! This is HTML5 Boilerplate.</p>
22
23    <script src="js/main.js"></script>
24  </body>
25 </html>
```

3- La vérification de la liste des commits sur github:

SaraMarhoum / Monsiteweb

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Branch: master

Commits on Nov 28, 2019

Renommer 404 to error404	1	Verified	2149260	<>
add balise h1 ds html	1	Verified	c0f275b	<>
add style file in css	1	Verified	ba6489e	<>

Commits on Nov 27, 2019

4- La vérification avec “git status” sur gitbush”:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/Monsiteweb-local (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

5- La vérification de repo local:

```
Youcode@DESKTOP-TLLL2MM MINGW64 ~/Desktop/projects/Monsiteweb-local (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

Explication et définition:

- 1- Git est un logiciel de gestion de versions décentralisé créé par Linus Torvalds. C'est un logiciel le plus populaire qui est utilisé par plus de douze millions de personnes.
- 2- Le git repository (ou history tree) est le fichier .git habituellement caché, il contient l'historique des commits du projet. Il garde en mémoire tous les changements qui ont été comités. C'est ce qui est cloné quand on clone la version d'un "serveur".
- 3- - Logs: endroit contenant les commits effectués et sa commande permet de les afficher.
 - Branches: endroit contenant les branches effectués
 - Head : le commit en cour où se trouve votre répo. Souvent le HEAD pointe vers le derniers commit dans notre branche en cour. En gros HEAD veut dire: vers quoi mon répo pointe actuellement.