**Install GIT & make sure it is added into PATH.**

**Use GIT as local VCS.**

1. **Create a directory ‘project\_dir’ & cd to ‘project\_dir’.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~

$ mkdir project\_dir

HP@LAPTOP-VH8V7PO3 MINGW64 ~

$ cd project\_dir

1. **Initialize git version database. (git init)**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir

$ git init

Initialized empty Git repository in C:/Users/HP/project\_dir/.git/

1. **Create a new file index.html.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ touch index.html

1. **Check the git status. You should find index.html as untracked file.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

No commits yet

Untracked files:

(use "git add <file>..." to include in what will be committed)

index.html

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

1. **Stage the index.html file.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git add index.html

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: index.html

1. **Commit index.html**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git commit -m "committing index file"

[master (root-commit) e9466ff] committing index file

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 index.html

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

nothing to commit, working tree clean

1. **Make few changes in index.html & create a new file info.txt file.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ vi index.html

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

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: index.html

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

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ touch info.txt

1. **Check git status. You should find index.html & info.txt as untracked files.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

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: index.html

Untracked files:

(use "git add <file>..." to include in what will be committed)

info.txt

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

1. **Configure GIT to ignore all txt files.**
2. HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)
3. $ touch .gitignore
4. **Again, check the git status. You should find only index.html as untracked file.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

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: index.html

1. **State & commit index.html**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git commit -a -m "committing updated index file"

warning: LF will be replaced by CRLF in index.html.

The file will have its original line endings in your working directory

[master 2d5f751] committing updated index file

1 file changed, 1 insertion(+)

1. **Log all your comments so far.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git log

commit 2d5f75121f3c9779df44d886b7723e679a853658 (HEAD -> master)

Author: Swathy Gopal N <swathygops@hotmail.com>

Date: Sun Feb 6 13:08:05 2022 +0530

committing updated index file

commit 31d1f7c13164e539f05b6b69e3829919956036cd

Author: Swathy Gopal N <swathygops@hotmail.com>

Date: Sun Feb 6 13:00:15 2022 +0530

committing index file

1. **Make some changes in index.html.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ vi index.html

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

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: index.html

Untracked files:

(use "git add <file>..." to include in what will be committed)

info.txt

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

1. **Revert the change made in the previous step using git command.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git restore index.html

1. **Again change index.html.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ vi index.html

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

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: index.html

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

1. **Stage index.html**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git add index.html

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

modified: index.html

1. **Revert back the last stage.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git restore --staged index.html

1. **Rename ‘add’ command to ‘my-add’.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git config --global alias.my-add add

1. **Using my\_add command Stage index.html again & commit the changes.**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git my-add index.html

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

modified: index.html

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git commit -m "committing index.html again"

[master 2c00e78] committing index.html again

1 file changed, 1 insertion(+), 1 deletion(-)

1. **Revert the last commit**

HP@LAPTOP-VH8V7PO3 MINGW64 ~/project\_dir (master)

$ git revert 2c00e7874cee340cd2f403bcef2e184a49cdf973

[master 804b4f3] Revert "committing index.html again"

1 file changed, 1 insertion(+), 1 deletion(-)

**GIT BRANCHING**

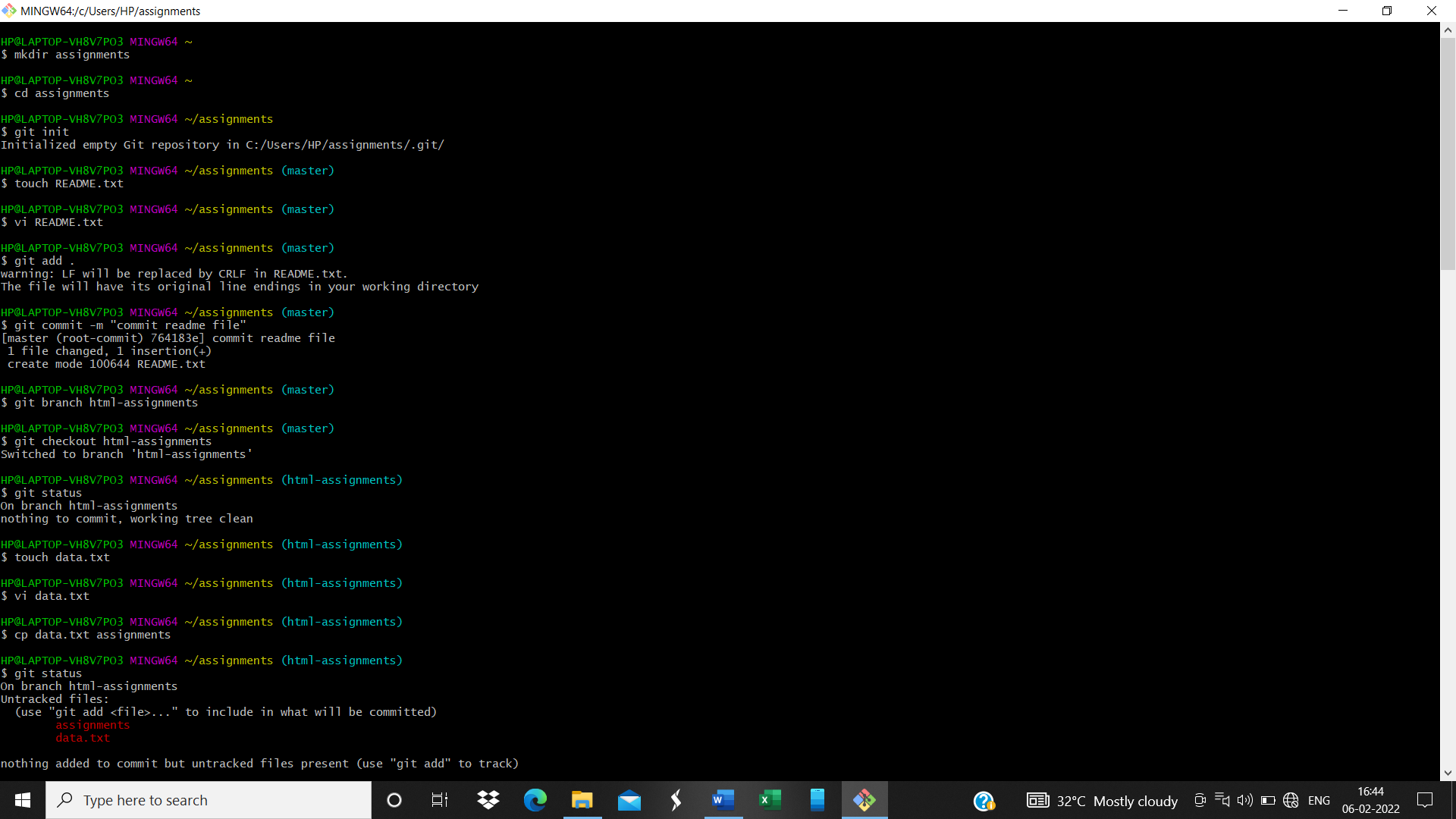
1. **First take a backup of your assignments & projects. This is required because due to incorrect GIT operation you may lose your files.**
2. **Create an empty directory ‘Assignments’ & cd to ‘Assignments’.**
3. **Create a file README.txt inside ‘Assignments’ & write few lines about the contents of ‘Assignments’ folder.**
4. **Commit README.txt file.**
5. **Now create a new branch ‘html-assignments’.**
6. **Switch to ‘html-assignments’ branch.**
7. **Copy all HTML assignments inside ‘Assignments’ folder.**
8. **Commit HTML assignments into ‘html-assignments’ branch.**
9. **Make minor changes into few files belonging to ‘html-assignments’ branch.**
10. **Commit those changed files.**
11. **Switch to master branch.**
12. **Make minor changes into README.txt file & commit those changes into master.**
13. **Again, switch to ‘html-assignments’ branch.**
14. **Make minor changes into few files belonging to ‘html-assignments’ branch.**
15. **Commit those changes.**
16. **Switch to master.**
17. **Merge ‘html-assignments’ branch into master. Confirm all html assignments are shown in master.**
18. **Finally delete the ‘html-assignments’ branch.**

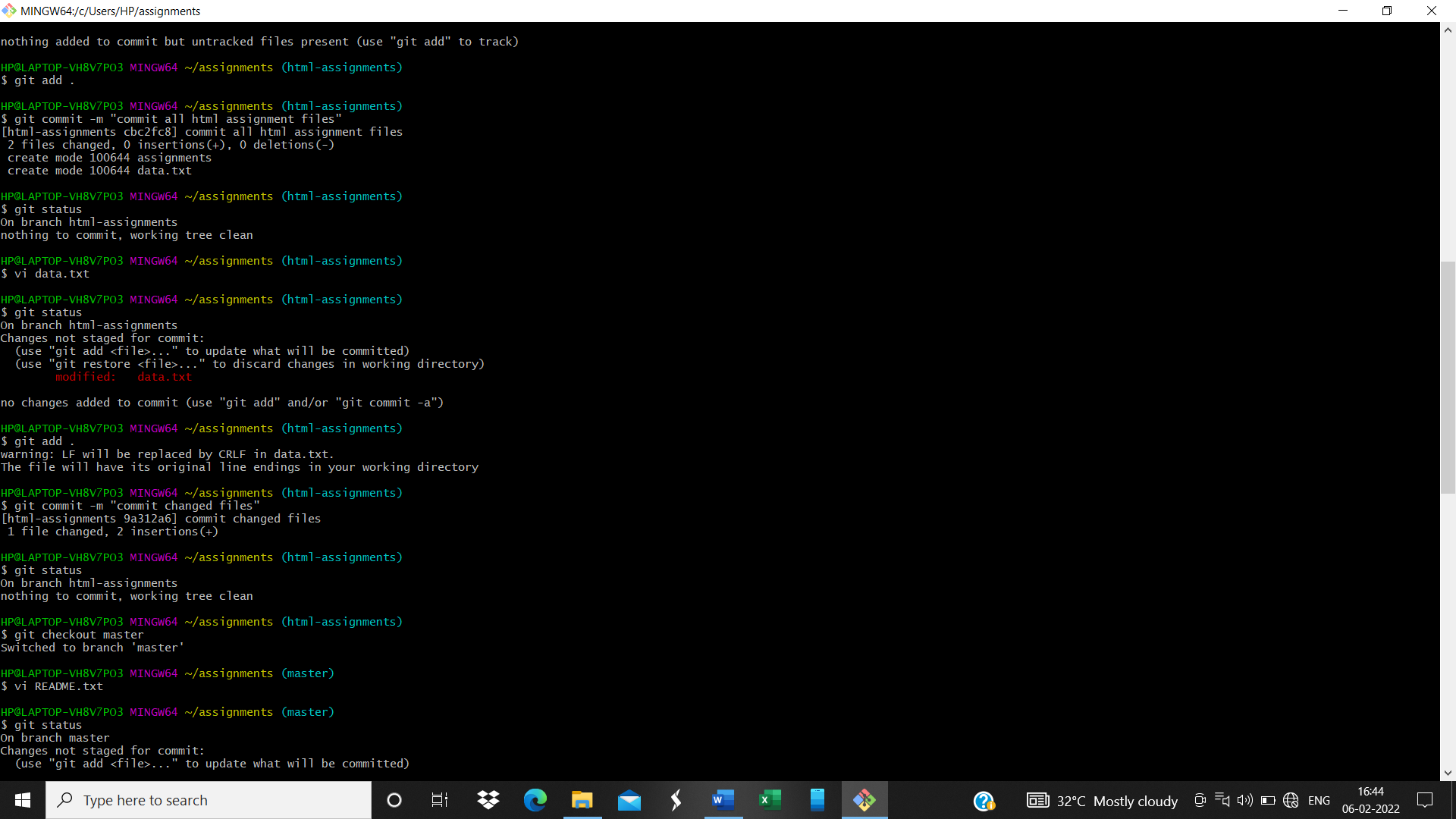
**SECTION-2 - (CSS assignments) Steps to follow:**

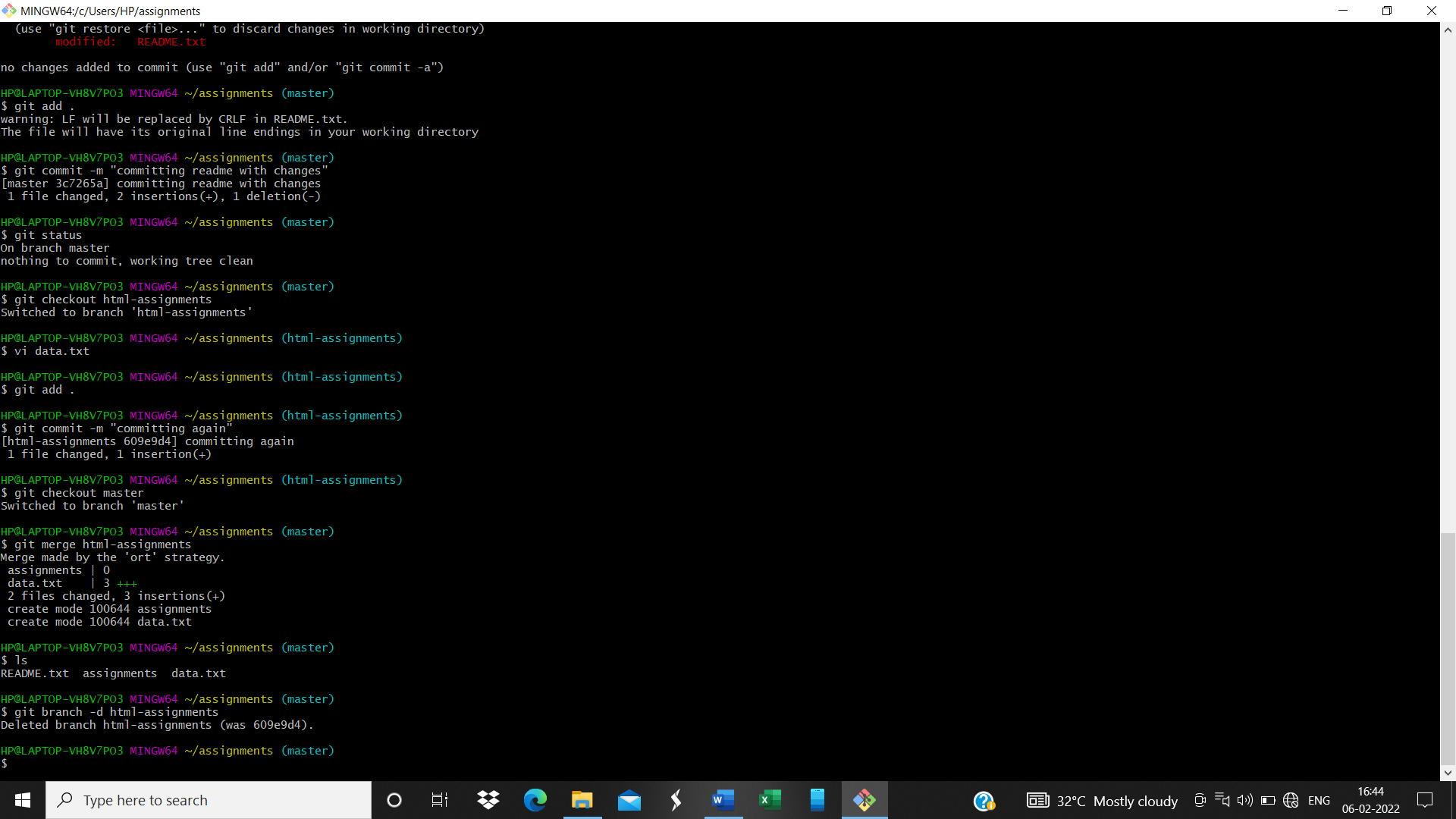
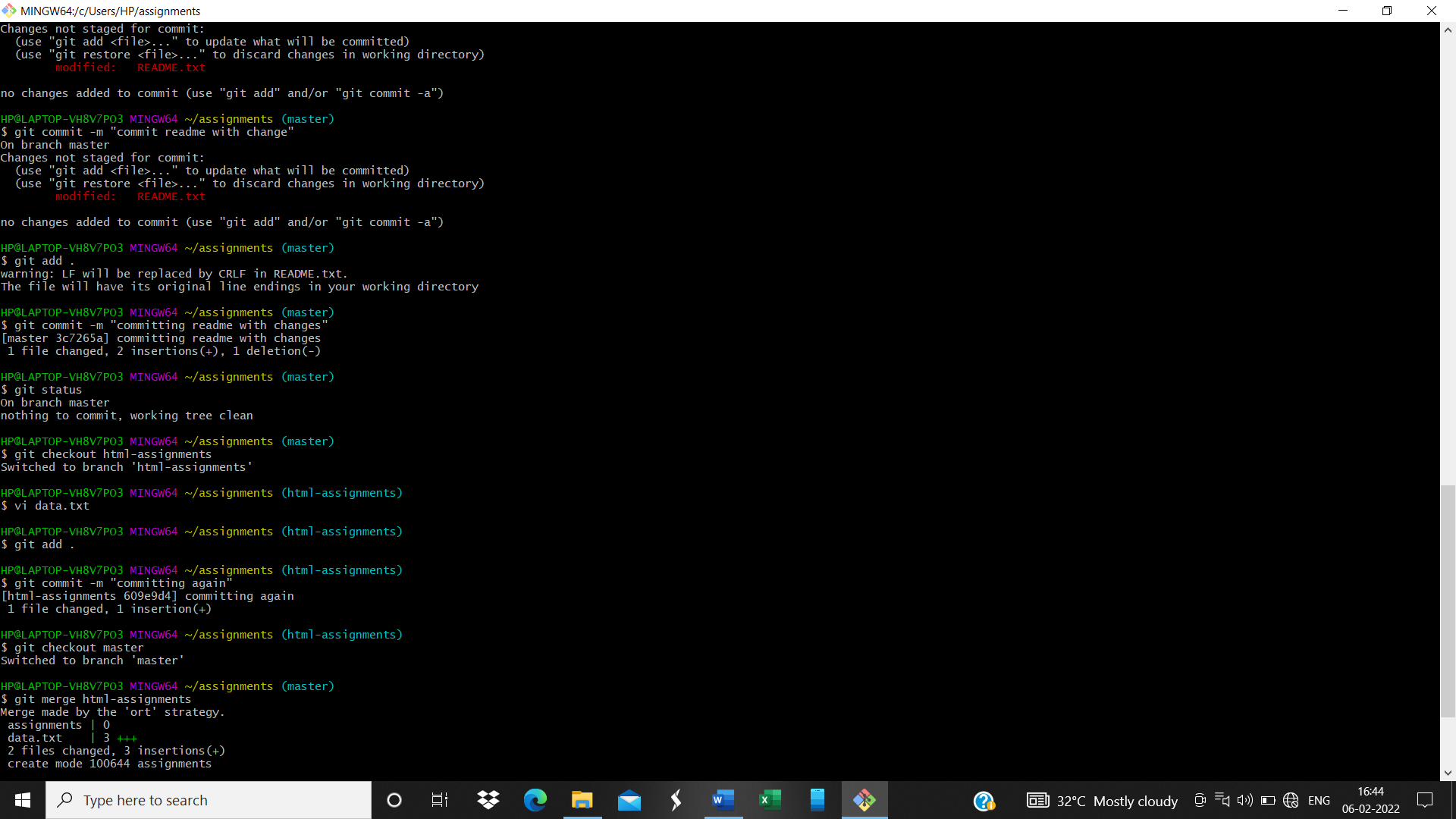
1. **Create a new branch ‘css-assignments’.**
2. **Switch to ‘css-assignments’ branch.**
3. **Copy all CSS assignments inside ‘Assignments’ folder.**
4. **Commit CSS assignments into ‘css-assignments’ branch.**
5. **Make minor changes into README.txt file on line 1 belonging to ‘css-assignments’ branch.**
6. **Commit those changed files.**
7. **Switch to master branch.**
8. **Make minor changes into README.txt file on line 3 & commit those changes into master.**
9. **Again, switch to ‘css-assignments’ branch.**
10. **Make minor changes into few files belonging to ‘css-assignments’ branch.**
11. **Commit those changes.**
12. **Switch to master.**
13. **Merge ‘css-assignments’ branch into master. Confirm all css assignments are shown in master.**
14. **Finally delete the ‘css-assignments’ branch.**

**SECTION-3 - (JavaScript assignments) Steps to follow:**

1. **Create a new branch ‘js-assignments’.**
2. **Switch to ‘js-assignments’ branch.**
3. **Copy all JavaScript assignments inside ‘Assignments’ folder.**
4. **Commit JavaScript assignments into ‘js-assignments’ branch.**
5. **Make minor changes into README.txt file on line 1 belonging to ‘js-assignments’ branch.**
6. **Commit those changed files.**
7. **Switch to master branch.**
8. **Make minor changes into README.txt file on line 1 & commit those changes into master.**
9. **Again, switch to ‘js-assignments’ branch.**
10. **Make minor changes into few files belonging to ‘js-assignments’ branch.**
11. **Commit those changes.**
12. **Switch to master.**
13. **Merge ‘js-assignments’ branch into master. Confirm all JavaScript assignments are shown in master.**
14. **Finally delete the ‘js-assignments’ branch.**

****

****

****