

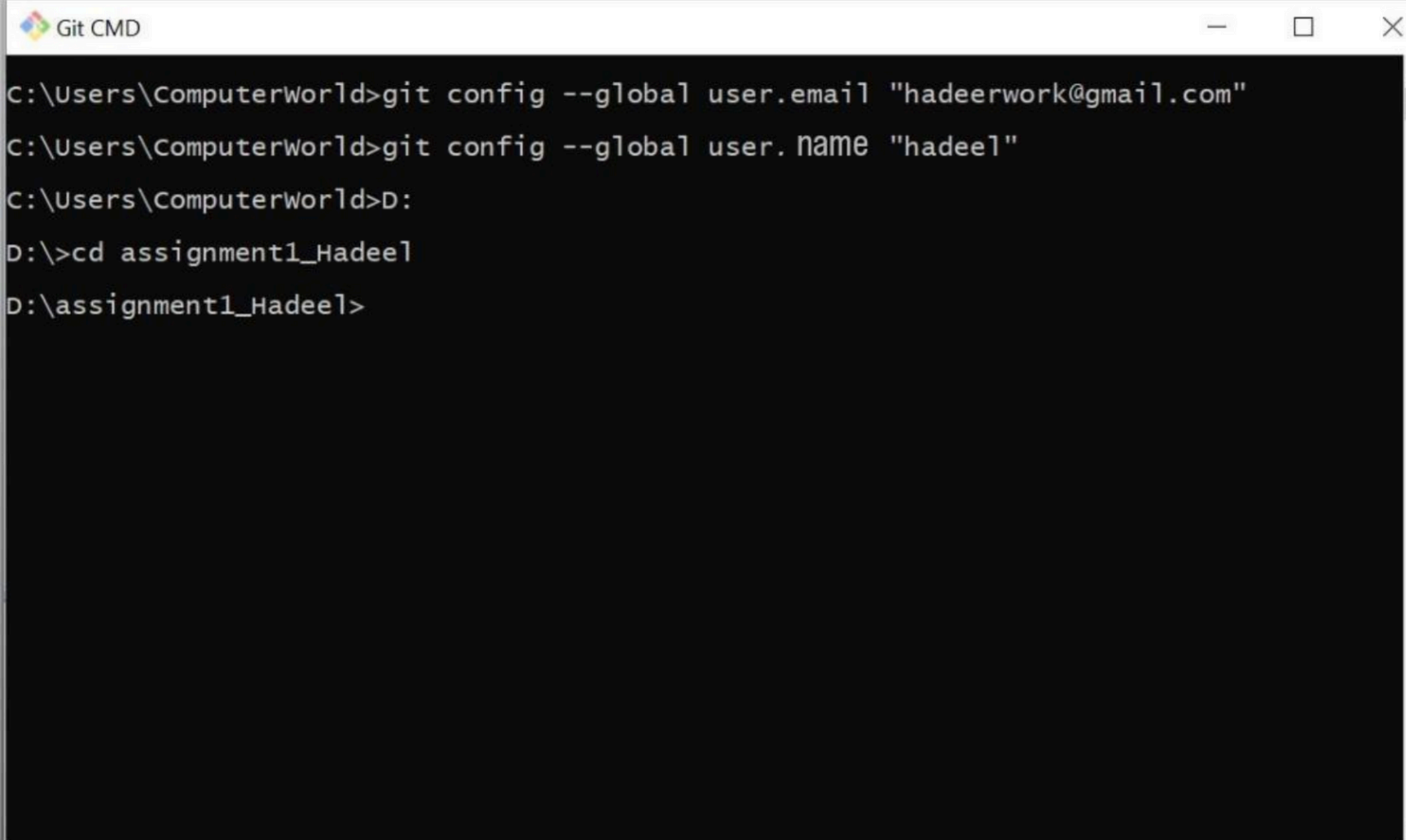


Software engineering

Assignment 1

By:Hadeel Al_homazh

Supervisor: eng.Malek Almosanif



```
Git CMD
C:\Users\ComputerWorld>git config --global user.email "hadeerwork@gmail.com"
C:\Users\ComputerWorld>git config --global user.name "hadeel"
C:\Users\ComputerWorld>D:
D:\>cd assignment1_Hadeel
D:\assignment1_Hadeel>
```

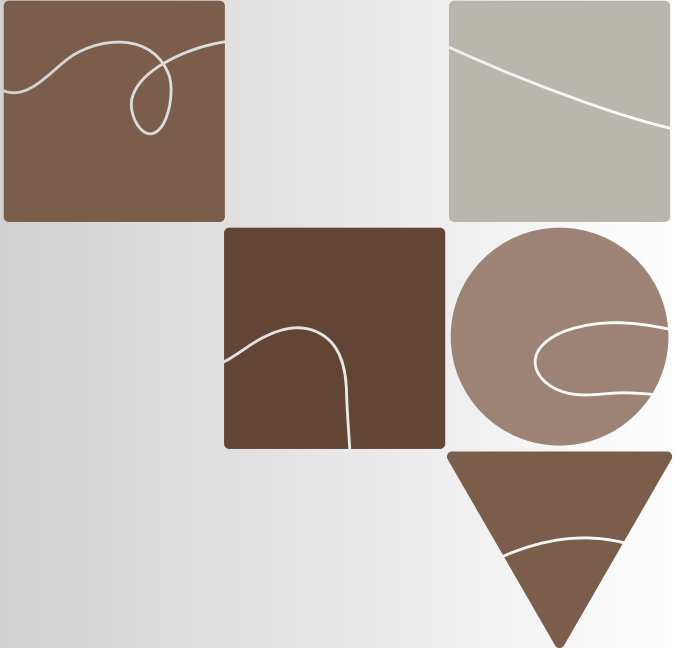
تم إعداد Git بتهيئة الاسم والبريد الإلكتروني الخاص
بالمستخدم باستخدام الأمرين:

`git config --global user.name`

`git config --global user.email`

لتسجيل معلومات المستخدم في كل عمليات الحفظ
(commit).

بعد ذلك تم الانتقال إلى القرص D و إلى مجلد المشروع
assignment1_Hadeel باستخدام الأمر **cd** للبدء بالعمل
داخل المجلد.



```
Select Git CMD
D:\assignment1_Hadeel>git init
Initialized empty Git repository in D:/assignment1_Hadeel/.git/

D:\assignment1_Hadeel>echo
ECHO is on.

D:\assignment1_Hadeel>echo "this is my first assignment" > first.txt

D:\assignment1_Hadeel>git status
On branch master

No commits yet

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

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

D:\assignment1_Hadeel>
```

تم إنشاء مستودع Git جديد داخل مجلد المشروع باستخدام الأمر **git init**

بعدها تم إنشاء ملف نصي جديد باسم **first.txt** يحتوي على عبارة "this is my first assignment" باستخدام أمر **echo**.

تم تنفيذ الأمر **git status** الذي أظهر حالة الملفات.



```
D:\assignment1_Hadeel>git add .

D:\assignment1_Hadeel>git commit --m "save to my first assignment"
[master (root-commit) ee2a592] save to my first assignment
1 file changed, 1 insertion(+)
create mode 100644 first.txt

D:\assignment1_Hadeel>notepad assignment1_Hadeel.txt

D:\assignment1_Hadeel>notepad first.txt

D:\assignment1_Hadeel>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:   first.txt

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

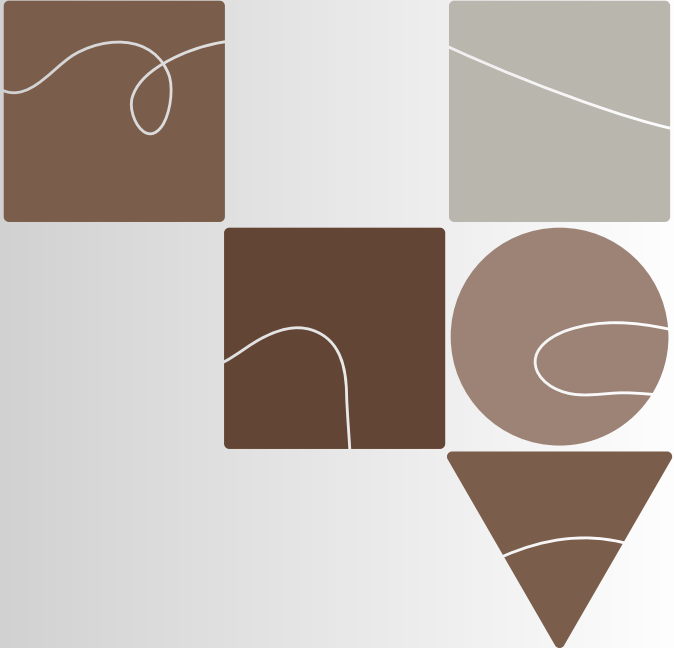
D:\assignment1_Hadeel>git add .

D:\assignment1_Hadeel>git commit -m "edit to my first assignment"
[master bc8fd25] edit to my first assignment
1 file changed, 1 insertion(+)
```

استخدمت الأمر **git add** لإضافة التغييرات إلى مرحلة staging ثم قمت بعمل أول حفظ رسمي للتغييرات باستخدام

git commit -m "save to my first assignment. "
مما أنشأ نسخة محفوظة من الملف **first.txt**.

بعد ذلك، فتحت الملف وعدلت عليه بواسطة المفكرة ثم نفذت **git status** الذي أظهر أن الملف قد تم تعديله ولكن لم يتم إضافته، فقامت مجددًا بإضافة التعديل باستخدام **git add** ثم حفظته بكوميت ثاني باستخدام **git commit -m "edit to my first assignment"** عشان تسجل التعديلات الجديدة.



```
D:\assignment1_Hadeel>git log
commit bc8fd255ae381cab5f3579dae1a49059cbfaa3c7 (HEAD -> master)
Author: Hadeer <hadeel>
Date:   Mon Jul 21 00:34:22 2025 +0300

    edit to my first assignment

commit ee2a592c5f124cb4896eb477bb5d9fd2c948c5ac
Author: Hadeer <hadeel>
Date:   Mon Jul 21 00:24:32 2025 +0300

    save to my first assignment

D:\assignment1_Hadeel> git checkout "bc8fd255ae381cab5f3579dae1a49059cbfaa3c7"
M       first.txt
Note: switching to 'bc8fd255ae381cab5f3579dae1a49059cbfaa3c7'.

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.

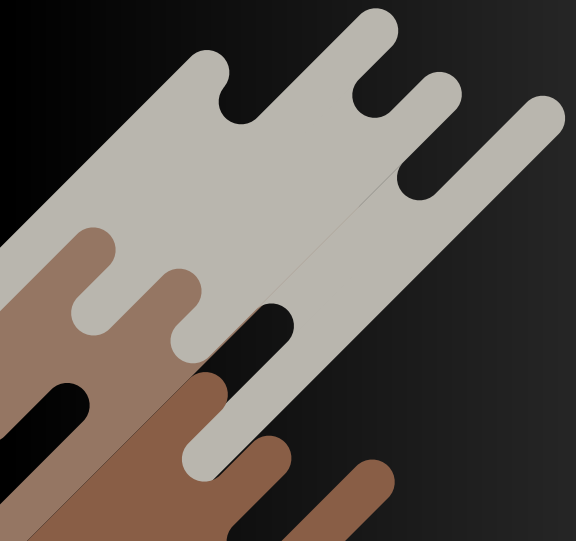
If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    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 bc8fd25 edit to my first assignment
D:\assignment1_Hadeel>
```



استخدمت الأمر **git log** لعرض سجل الحفظ (commit history) حيث ظهر لي كومتين محفوظين باسم المستخدم وهما:
"save to my first assignment"
و"edit to my first assignment"

ثم قمت بتنفيذ أمر **git checkout** للانتقال إلى الكوميت الثاني باستخدام الـ id الخاص به، مما وضع المشروع في حالة "detached HEAD" التي تسمح بمراجعة الكود في تلك المرحلة دون التأثير على الفرع الرئيسي الي هو الماستر.

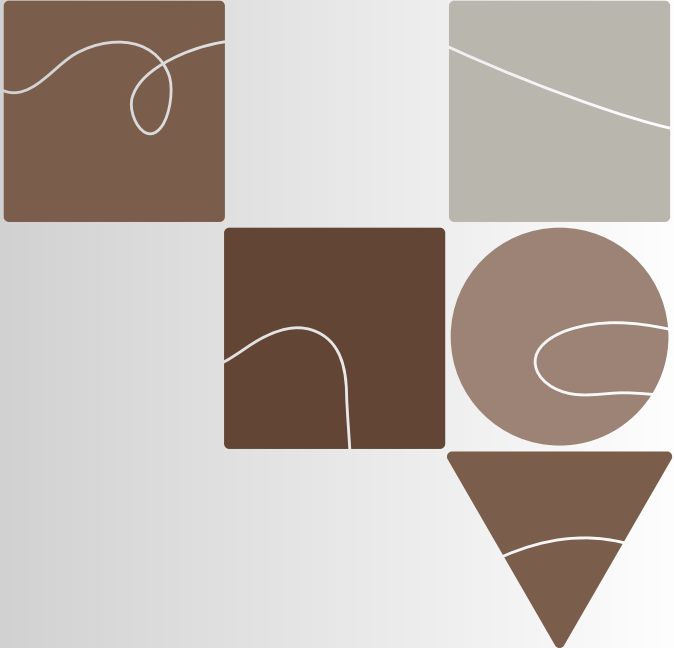
```
D:\assignment1_Hadeel>git branch
* (HEAD detached at bc8fd25)
master
```

```
D:\assignment1_Hadeel>git branch my_branch
```

```
D:\assignment1_Hadeel>_
```

أنشأت الفرع my_branch باستخدام **git branch**
من الكوميت الثاني،

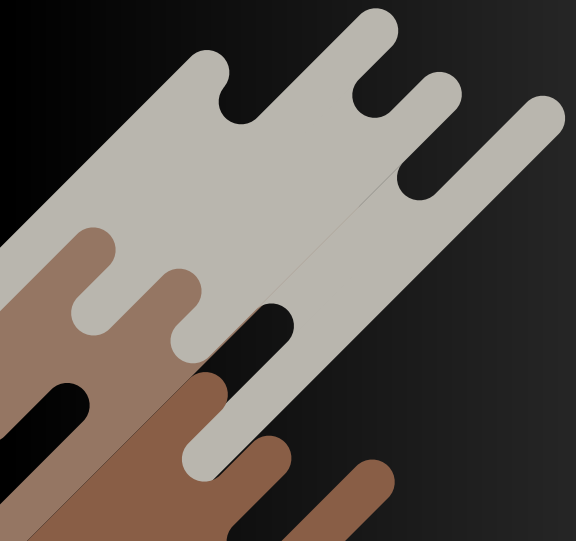
بحيث اقدر ابدأ تعديلات عليه بدون التأثير على الفرع
الأساسي master.



```
D:\assignment1_Hadeel>echo "this is my second assignment" > second.txt
D:\assignment1_Hadeel>git switch my_branch
M      first.txt
Switched to branch 'my_branch'
D:\assignment1_Hadeel> git checkout my_branch
M      first.txt
Already on 'my_branch'
D:\assignment1_Hadeel>git status
On branch my_branch
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:   first.txt

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

no changes added to commit (use "git add" and/or "git commit -a")
D:\assignment1_Hadeel>git switch master
M      first.txt
Switched to branch 'master'
D:\assignment1_Hadeel>_
```



أنشأت ملف جديد باسم `second.txt` يحتوي على عبارة `"this is my second assignment"`
ثم انتقلت إلى الفرع الجديد `my_branch` باستخدام أمر `git switch` وتأكدت ذلك بأمر `git checkout`،
وبعدها استخدمت `git status` لتفقد حالة .

بعدها عدت للفرع الأساسي `master` باستخدام أمر `git switch master`.

```
D:\assignment1_Hadeel>git merge my_branch
Already up to date.
```

```
D:\assignment1_Hadeel>git ls-files
first.txt
```

```
D:\assignment1_Hadeel>git clean -dn
Would remove second.txt
```

```
D:\assignment1_Hadeel>git clean -df
Removing second.txt
```

```
D:\assignment1_Hadeel>git restore first.txt
```

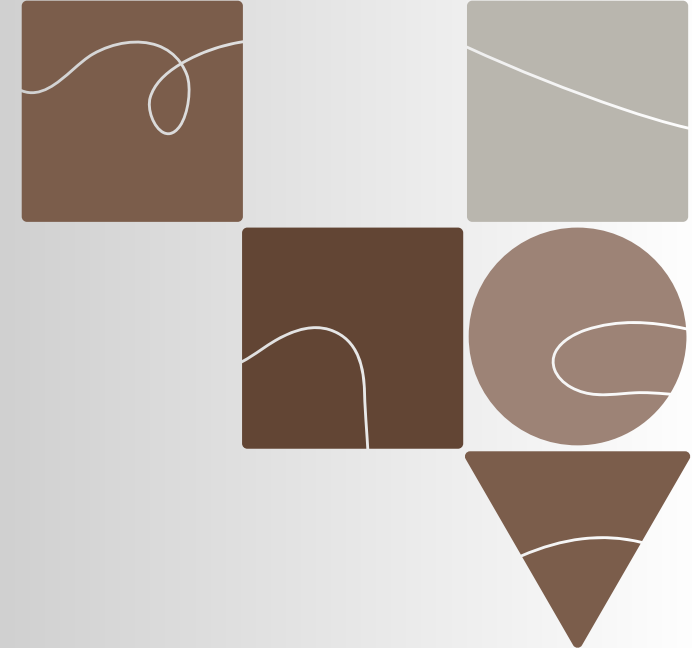
```
D:\assignment1_Hadeel>git restore .
```

```
D:\assignment1_Hadeel>git rm first.txt
rm 'first.txt'
```

```
D:\assignment1_Hadeel>git reset first.txt
fatal: ambiguous argument 'first.txt': unknown revision or path not in the working tree.
Use '--' to separate paths from revisions, like this:
'git <command> [<revision>...] -- [<file>...]'
```

```
D:\assignment1_Hadeel>git add first.txt
fatal: pathspec 'first.txt' did not match any files
```

هنا دمجت الفرع my_branch في master باستخدام **git merge** لكن لم يحدث أي تغيير لأن الفرعين محدثين. ب
بعدها استخدمت **git clean -dn** لمعاينة الملفات التي سيتم حذفها غوحدت أن second.txt سيتم حذفه، ثم حذفته فعليًا باستخدام **git clean -df**. بعد ذلك استخدمت **git restore** لاستعادة حالة الملف first.txt، ثم حذفته باستخدام **git rm**. وبعدها حاولت إعادة تعقبه بـ **git add** لكنه لم يُعثر عليه لأنه كان قد تم حذفه سابقًا.



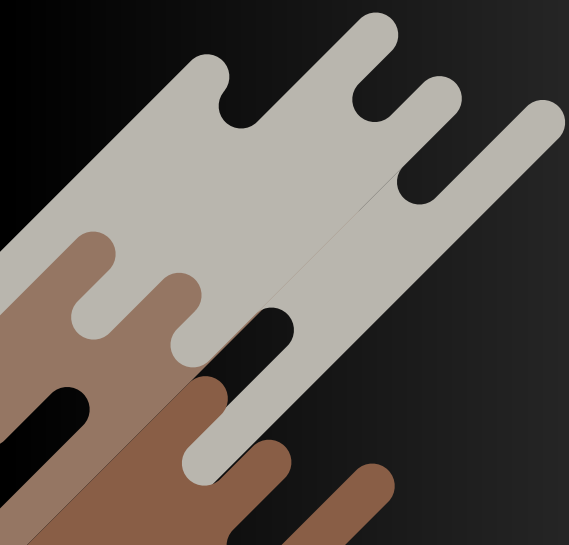
```
D:\assignment1_Hadeel>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    first.txt

D:\assignment1_Hadeel>git restore --staged first.txt
D:\assignment1_Hadeel>git restore first.txt
D:\assignment1_Hadeel>
```

في هذه الخطوة، كان الملف `first.txt` محذوفًا لكنه لا يزال في مرحلة الـ `staging` (جاهز للإرسال في `commit`).

لذلك، قمت أولاً بإلغاء تتبع الحذف باستخدام الأمر `git restore --staged first.txt` لإخراجه من الـ `staging`.

ثم استخدمت `git restore first.txt` لاستعادة الملف فعليًا إلى مجلد المشروع من سجل `Git`، وبذلك تم التراجع تمامًا عن الحذف.





Thanks