

需要先建立一個倉庫(repository)。使用一個目錄作為 Git 倉庫或創建一個空目錄。

使用目前的目錄作為 Git 倉庫，初始化。

```
git init
```

使用我們指定目錄作為 Git 倉庫。

```
git init newrepo
```

假設在 Git 倉庫根目錄下。

添加新文件

目前倉庫，但什麼也沒有，可以使用 add 命令添加檔。

```
git add filename
```

可以使用 add... 繼續添加任務檔。

提交版本

目前添加了這些檔，希望它們能夠真正被保存在 Git 倉庫。

提交到倉庫。

```
git commit -m "Adding files"
```

如果您不使用-m，會出現編輯器來讓你寫自己的注釋資訊。

當我們修改了很多檔，而不想每一個都 add，想 commit 自動來提交本地修改，我們可以使用-a 標識。

```
git commit -a -m "Changed some files"
```

git commit 命令的-a 選項可將所有被修改或者已刪除的且已經被 git 管理的文檔提交到倉庫中。

(1)\$ git add -f --all

("檔案" "的改變" 強制加入版本管理)

查看目前 git 的記錄和目標檔案的內容有沒有差異，有的話就會加進去。

git add 並不會直接寫入 git 的記錄檔，會被另外記錄起來. git commit 才一次執行寫入所有的指令。

(2)\$ git commit -m "<xxxxx>" 。

\$ git commit -m " addnewfiletxt"

\$ git log

可以看到所有的紀錄，每條紀錄都會有一個編碼(版號)。

\$ git show 後面加上前四碼(或四碼以上)這個指令

\$ git show w56e

顯示出那一條紀錄的變動

回到某一版

```
$ git reset --hard <版號>
```

```
$ git reset --hard 2d3bde
```

git log 去看，紀錄變少了!!

未來版本

git log 其實只能看到過去的版本

```
$ git reflog
```

```
$ git reset --hard <版號>
```

```
$ git reset --hard d53d4e
```