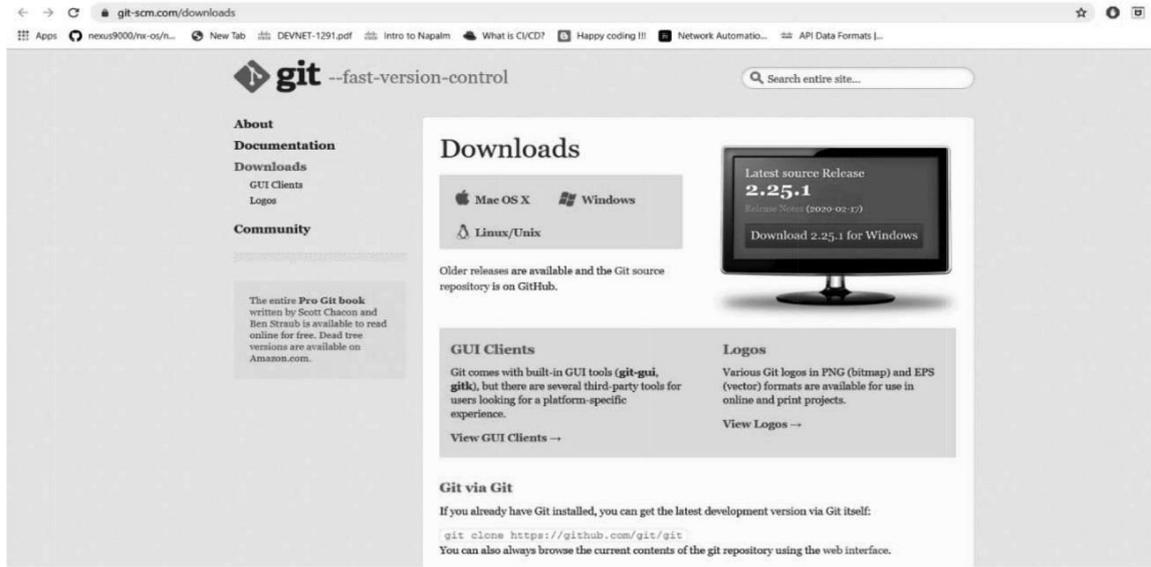


## LAB - CÁC THAO TÁC CƠ BẢN TRÊN GIT

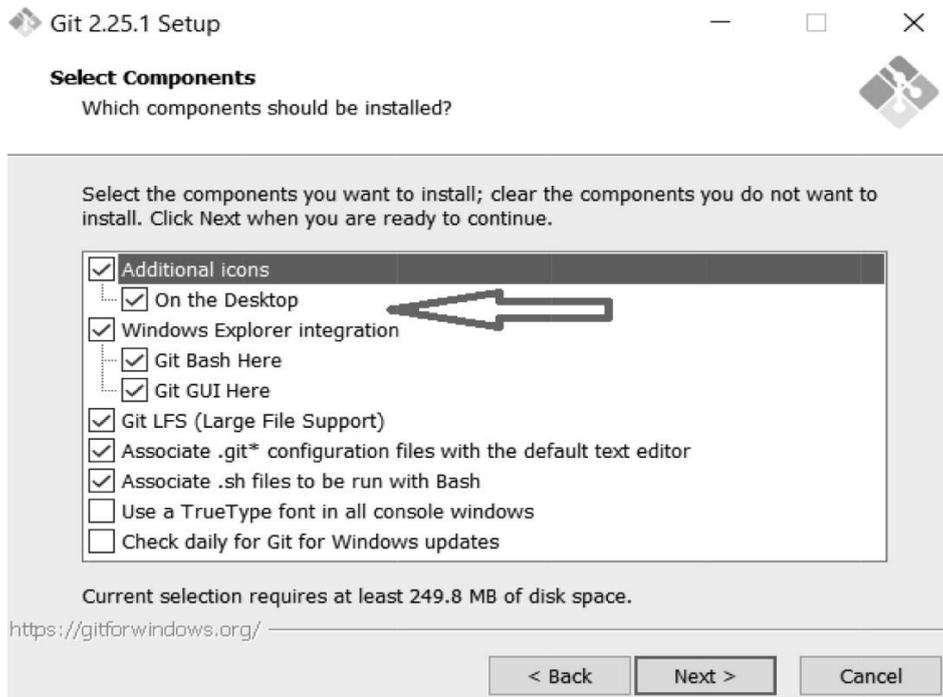
### I. Version Control

Cài đặt Git cho máy tính, ở đây sử dụng Windows.

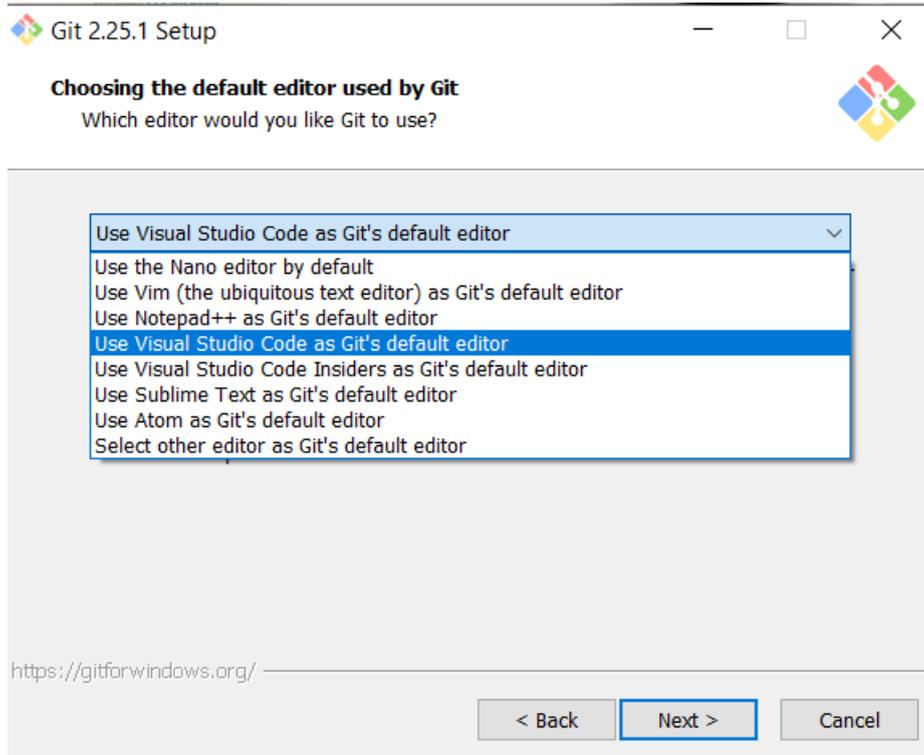
Vào đường dẫn <https://git-scm.com/downloads> để tiến hành download Git:



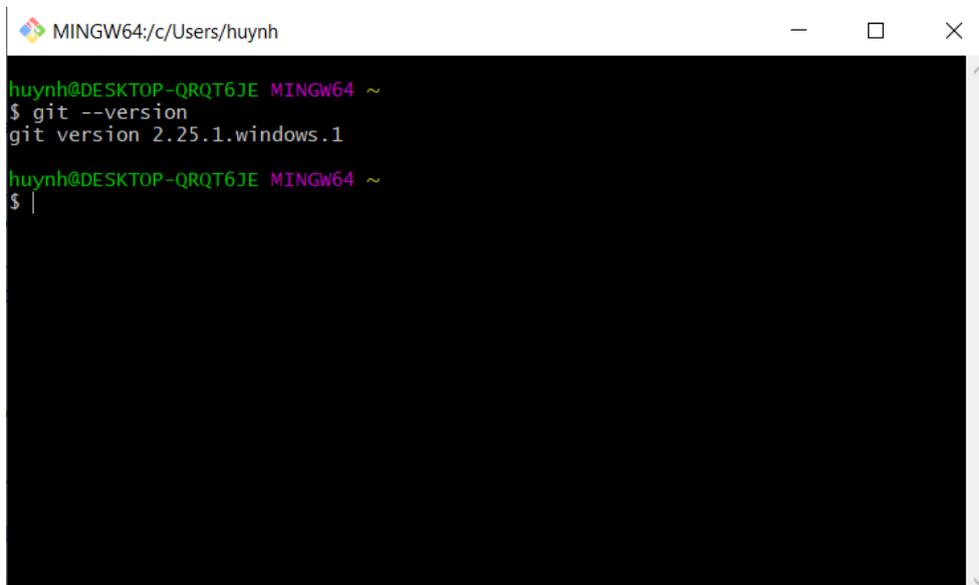
Chạy file vừa tải vào Next, sau đó đánh dấu tích như trong hình:



Ở mục này chúng ta có thể chọn Git's default editor, tùy vào phần mềm code của bạn để chọn mục này. Sau đó Next đến hết rồi Install là đã hoàn tất việc cài đặt Git:

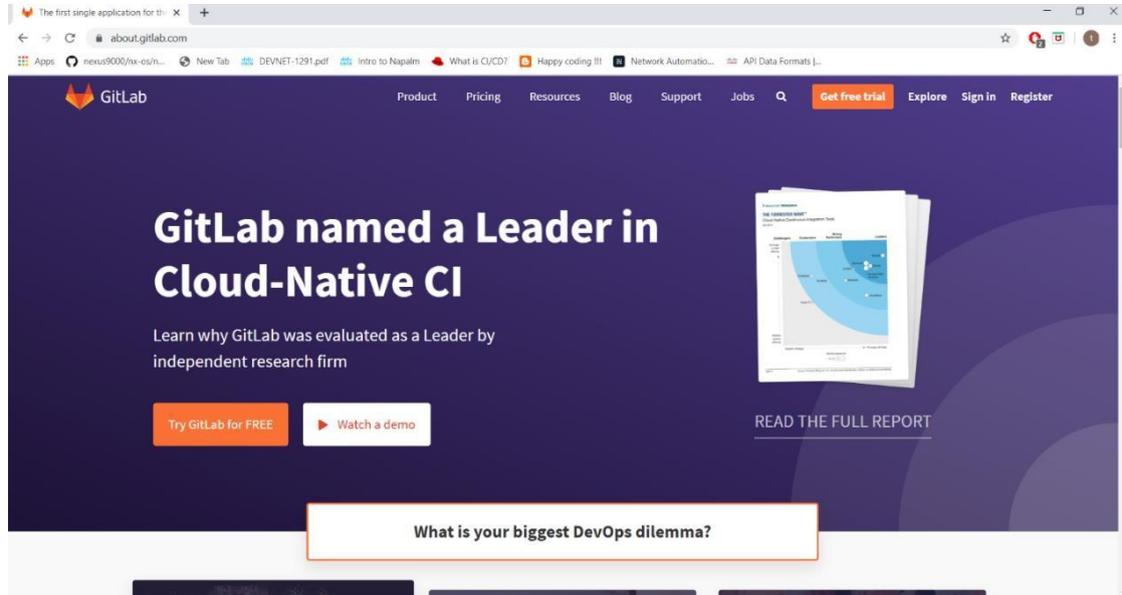


Để kiểm tra việc cài đặt có thành công hay không chúng ta vào Git Bash và nhập lệnh `git --version`:

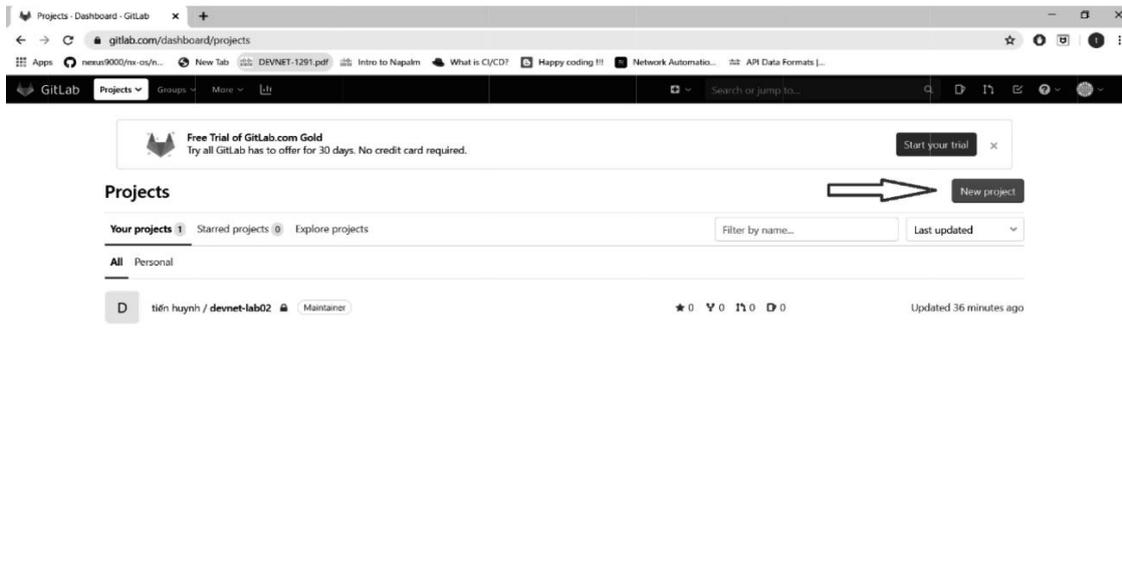


```
MINGW64:/c/Users/huynh
huynh@DESKTOP-QRQT6JE MINGW64 ~
$ git --version
git version 2.25.1.windows.1
huynh@DESKTOP-QRQT6JE MINGW64 ~
$ |
```

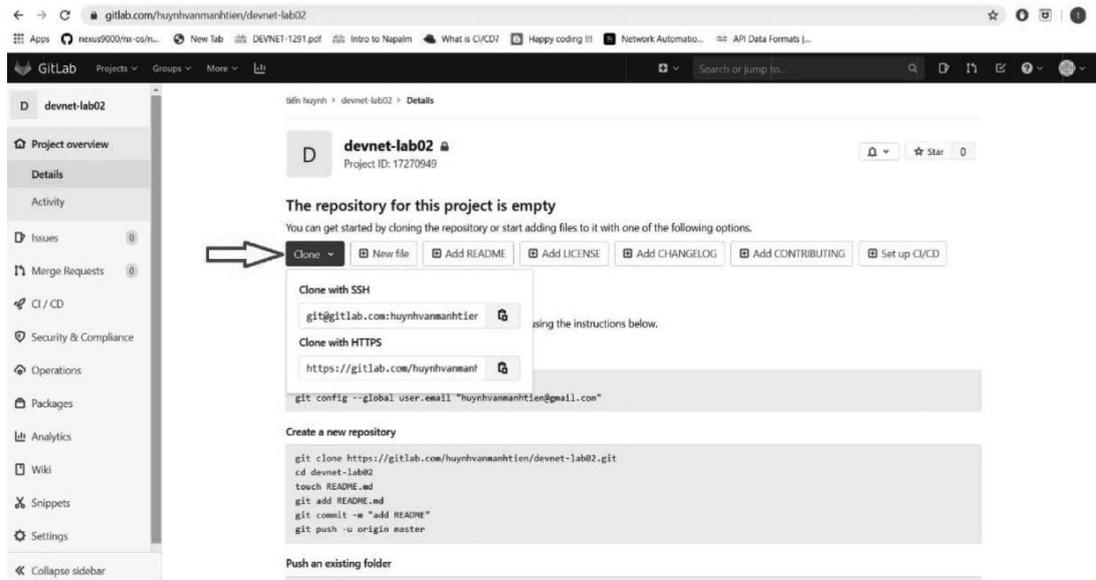
Mở trình duyệt web và truy cập vào đường dẫn <http://gitlab.com>. Chọn Sign in để đăng nhập vào Gitlab:



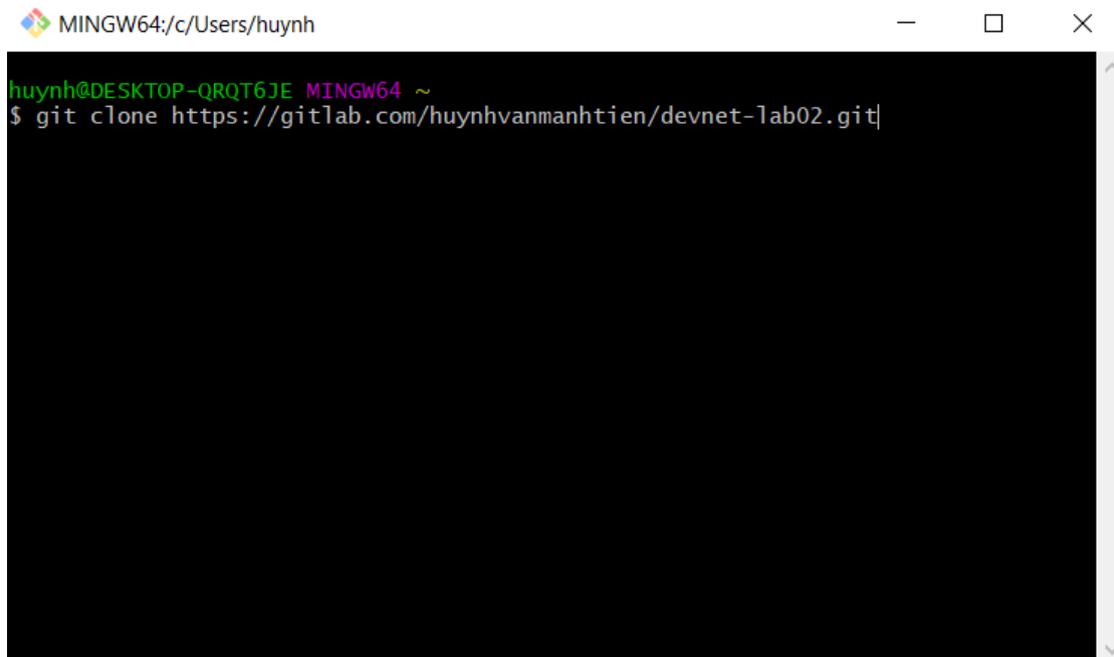
Chọn New project để tiến hành tạo 1 project mới. Sau khi tạo thành công, project vừa tạo sẽ xuất hiện ở mục Your project. Tiếp theo bạn sẽ clone repository này về máy tính của mình:



Click vào project vừa tạo, chọn Clone và click vào Copy icon để lấy đường link dùng để clone project:



Mở Git Bash nhập lệnh `git clone` và đường dẫn mới lấy được, sau đó nhập username và password đăng nhập vào Gitlab của bạn để xác thực:



Sử dụng câu lệnh `cd devnet-lab02` (trong đó `devnet-lab02` là tên project các bạn đặt lúc này) để vào thư mục vừa clone về. Sử dụng lệnh `ls -la` để show lên danh sách các thư mục hiện đang có trong project, ta sẽ thấy thư mục `README.md` ở trên Gitlab đã được download về máy tính của bạn:

```
MINGW64/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~
$ cd devnet-lab02/

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ ls -la
total 28
drwxr-xr-x 1 huynh 197609 0 Mar  4 09:47 ./
drwxr-xr-x 1 huynh 197609 0 Mar  4 09:43 ../
drwxr-xr-x 1 huynh 197609 0 Mar  4 09:46 .git/
-rw-r--r-- 1 huynh 197609 0 Mar  4 09:47 README.md

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

## II. Commit and Push Changes

Ở mục này bạn sẽ học cách xác định trạng thái của local repository, cách để add files, commit files và sau đó là push files lên remote repository.

Mở Git Bash và tạo 1 file có tên là lab02.py bằng cách sử dụng câu lệnh *touch*:

```
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ touch lab02.py

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ ls
lab02.py  README.md
```

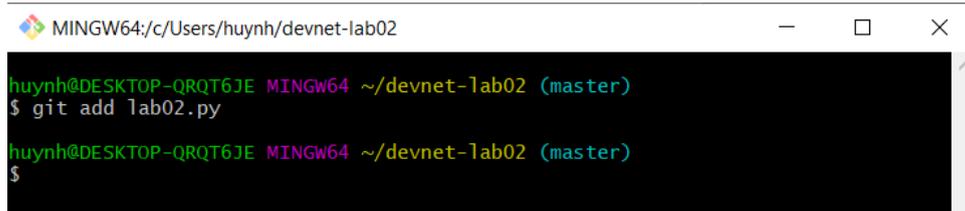
Chạy câu lệnh *git status* để kiểm tra trạng thái hiện tại của repository:

```
MINGW64/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  lab02.py

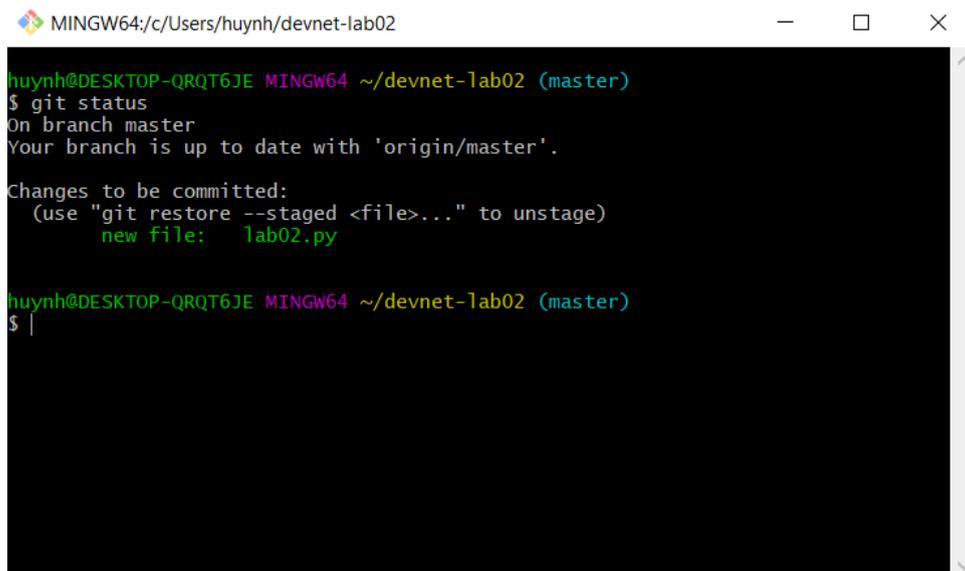
nothing added to commit but untracked files present (use "git add" to track)
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$
```

Để add file lab02.py lên repository, trước tiên bạn cần đánh dấu nó bằng câu lệnh `git add lab02.py`:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git add lab02.py
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$
```

Chạy lại lệnh `git status`, ta sẽ thấy file lab02.py bây giờ đã được đánh dấu và sẽ được đẩy lên repository trong lần commit tiếp theo:

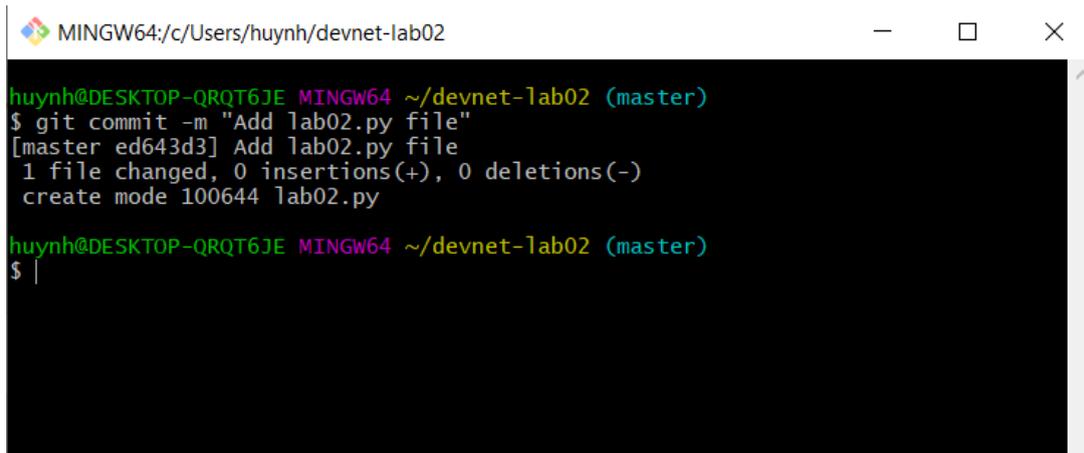


```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   lab02.py

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

Commit file vào local repository bằng câu lệnh `git commit -m`. Nên thêm chú thích vào mỗi lần commit (ở trong hình đó là câu lệnh “Add lab02.py file”):



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git commit -m "Add lab02.py file"
[master ed643d3] Add lab02.py file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 lab02.py

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

Chạy lại lệnh `git status` sau khi commit bạn sẽ thấy bản sao của nhánh master sẽ được đẩy vào `origin/master` trên Gitlab server trong lần commit tiếp theo. Có nghĩa là file `lab02.py` hiện tại vẫn là 1 phần của local repository nhưng chưa có trên Gitlab remote server:

```
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

Để publish những lần commit ta sử dụng lệnh `git push origin master`. Nhánh local master của bạn sẽ đồng bộ hóa với nhánh master của remote:

```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 263 bytes | 263.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0)
To https://gitlab.com/huynhvmanhtien/devnet-lab02.git
   0fd683f..ed643d3  master -> master

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

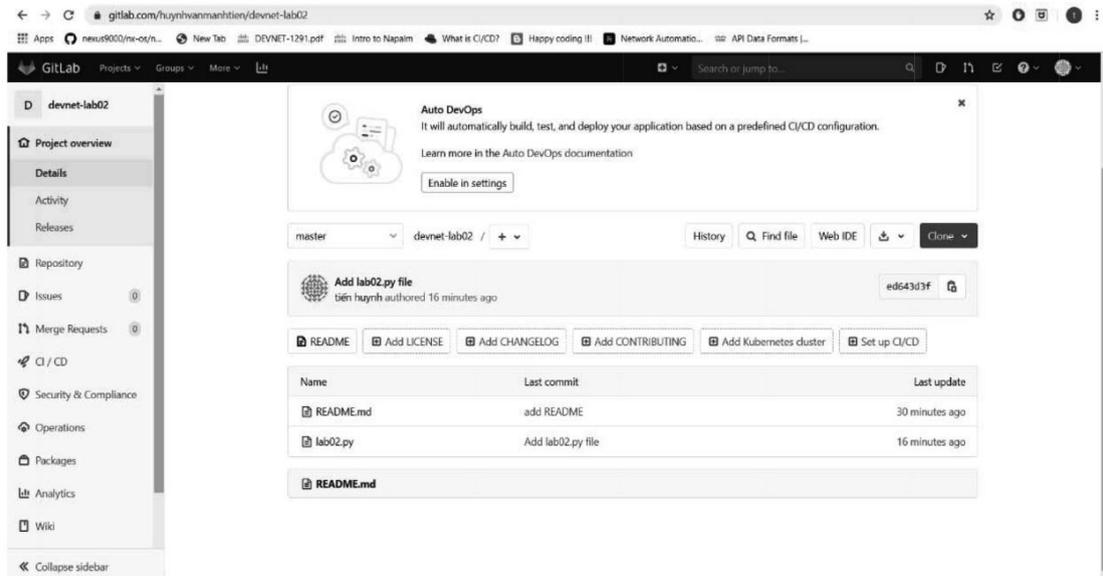
Chạy lệnh `git status` để kiểm tra:

```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

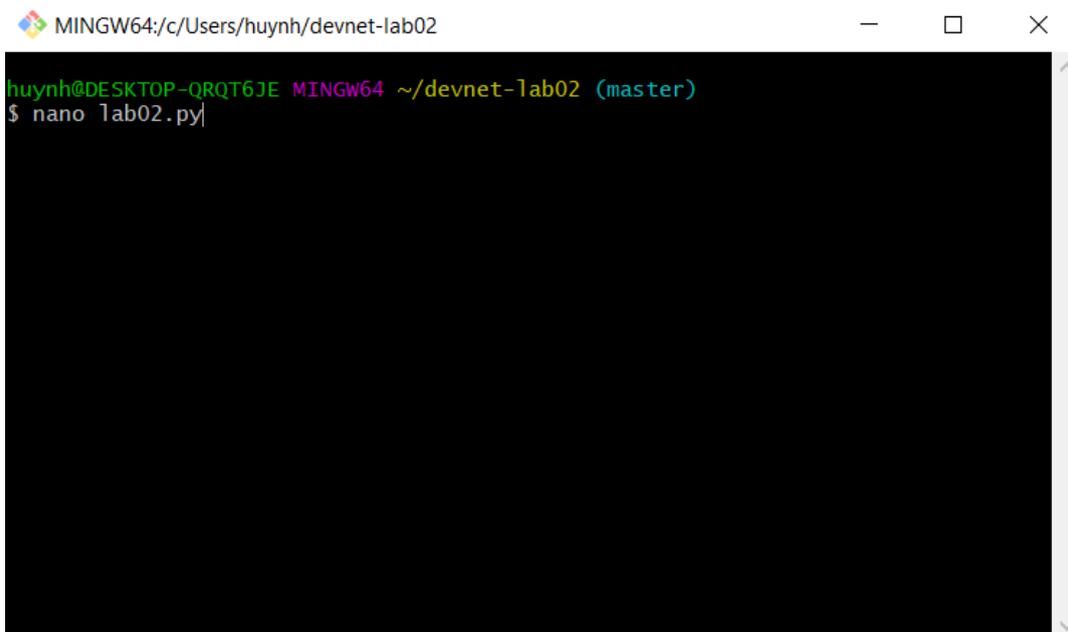
nothing to commit, working tree clean

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

Lúc này vào Gitlab, vào project mà bạn tạo sẽ thấy file lab02.py đã được đẩy lên remote repository:



Thêm một đoạn code vào dòng lệnh vào file lab02.py để tiếp tục kiểm tra. Sửa file bằng câu lệnh `nano lab02.py`:

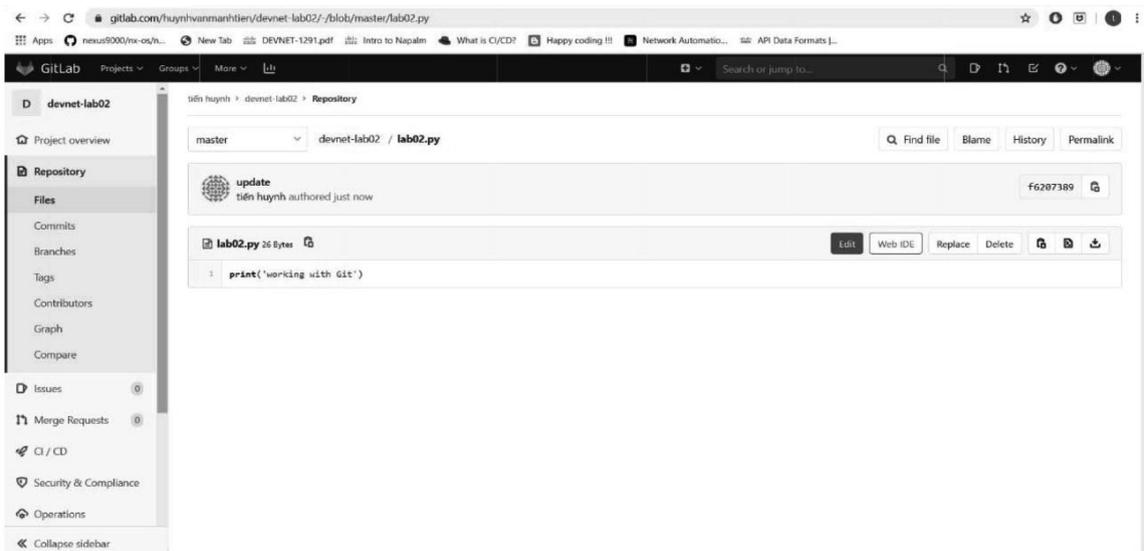


Ở đây mình thêm vào file lab02 một đoạn code `print('working with Git')`:

```
MINGW64:/c:/Users/huynh/devnet-lab02
GNU nano 4.8 lab02.py
print('working with Git')

[ Read 1 line ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line
```

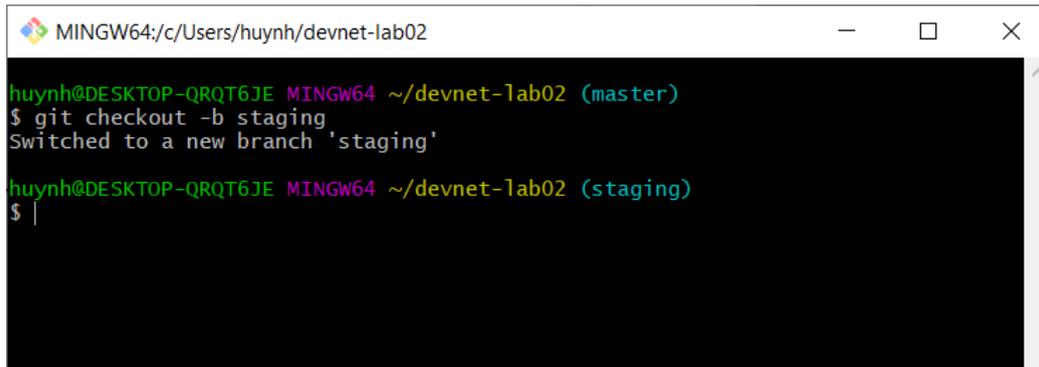
Sau đó tiếp tục dùng lệnh `git add`, `git commit -m` và `git push origin master` để update file lab02.py vừa chỉnh sửa sẽ được kết quả như trong hình:



### III. Compare Branches

Ở phần này bạn sẽ học cách tạo 1 branch, di chuyển qua lại giữa các branch và cách đối chiếu các branch để tìm các file khác nhau.

Mở Git Bash, tạo một branch mới trên local repository tới tên là “*staging*” và chuyển qua nhánh mới bằng câu lệnh `git checkout -b staging`:

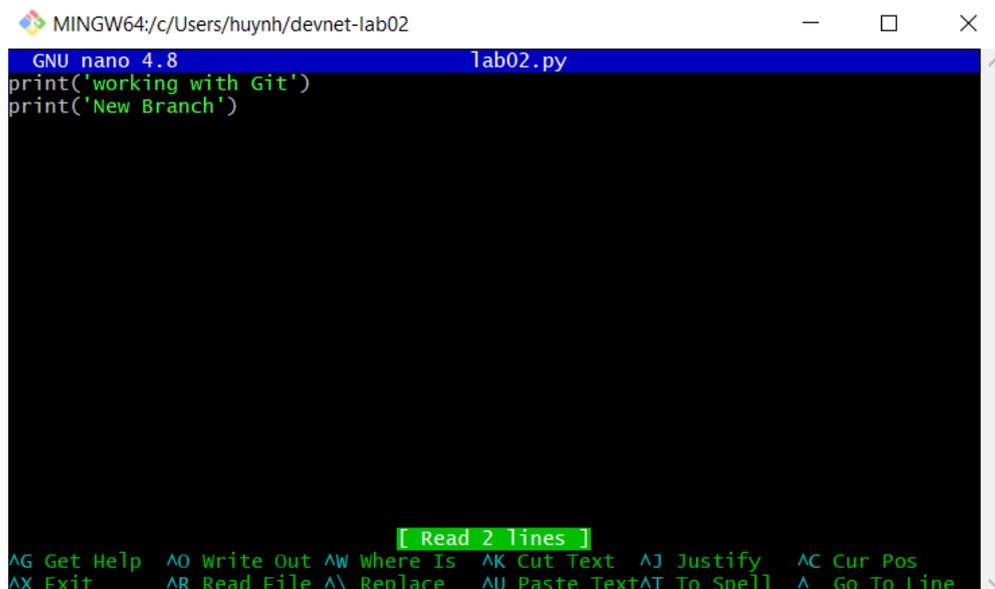


```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git checkout -b staging
Switched to a new branch 'staging'
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ |
```



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git checkout -b staging
Switched to a new branch 'staging'
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ nano lab02.py
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ |
```

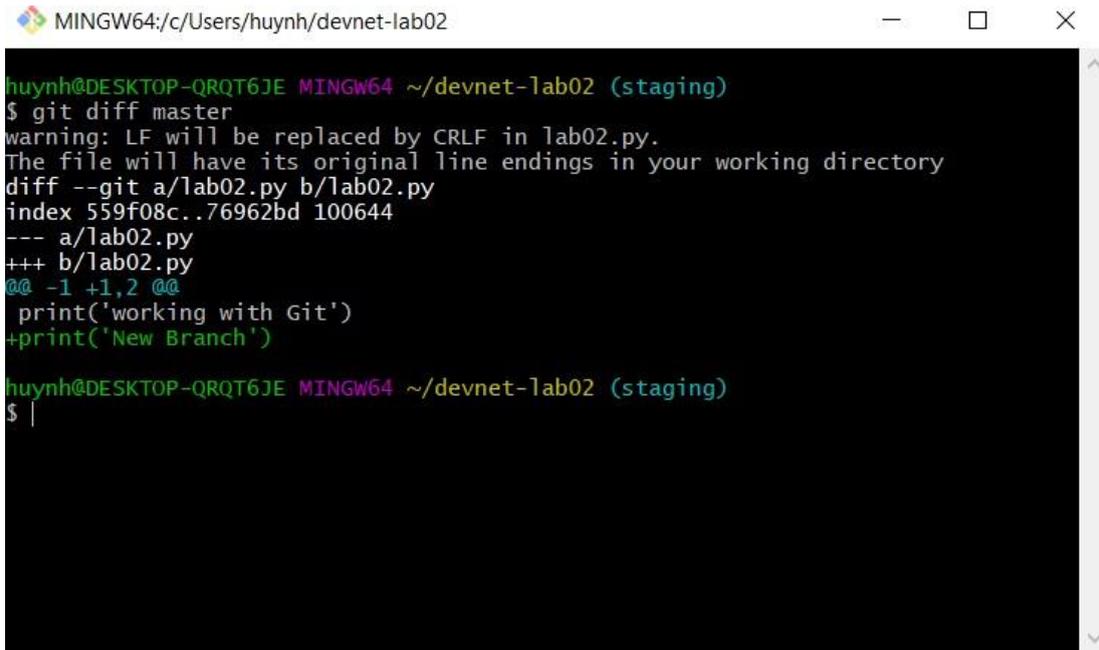
Vào file `lab02.py` sau đó thêm một dòng `print('New Branch')`:



```
MINGW64:/c/Users/huynh/devnet-lab02
GNU nano 4.8 lab02.py
print('working with Git')
print('New Branch')

[ Read 2 lines ]
AG Get Help  AO Write Out  AW Where Is  AK Cut Text  AJ Justify  AC Cur Pos
AX Exit     AR Read File  A\ Replace  AU Paste TextAT To Spell  A_ Go To Line
```

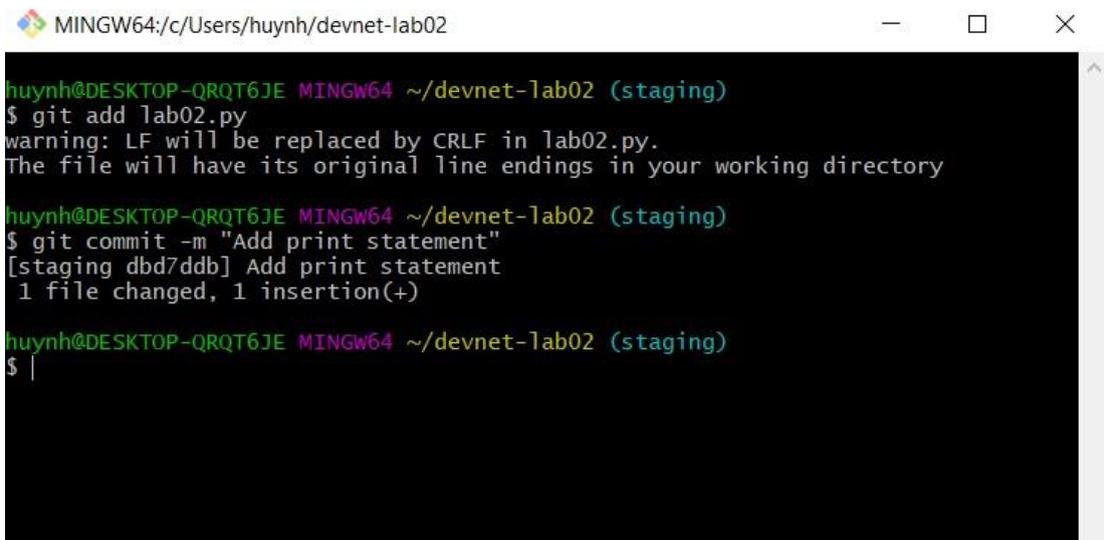
Sau đó ta tiến hành đối chiếu branch vừa tạo với branch master bằng câu lệnh *git diff master*. Branch vừa tạo có 2 dòng print trong khi branch master chỉ có 1:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ git diff master
warning: LF will be replaced by CRLF in lab02.py.
The file will have its original line endings in your working directory
diff --git a/lab02.py b/lab02.py
index 559f08c..76962bd 100644
--- a/lab02.py
+++ b/lab02.py
@@ -1,2 @@
 print('working with Git')
+print('New Branch')

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ |
```

Add file lab02.py để đánh dấu và commit:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ git add lab02.py
warning: LF will be replaced by CRLF in lab02.py.
The file will have its original line endings in your working directory

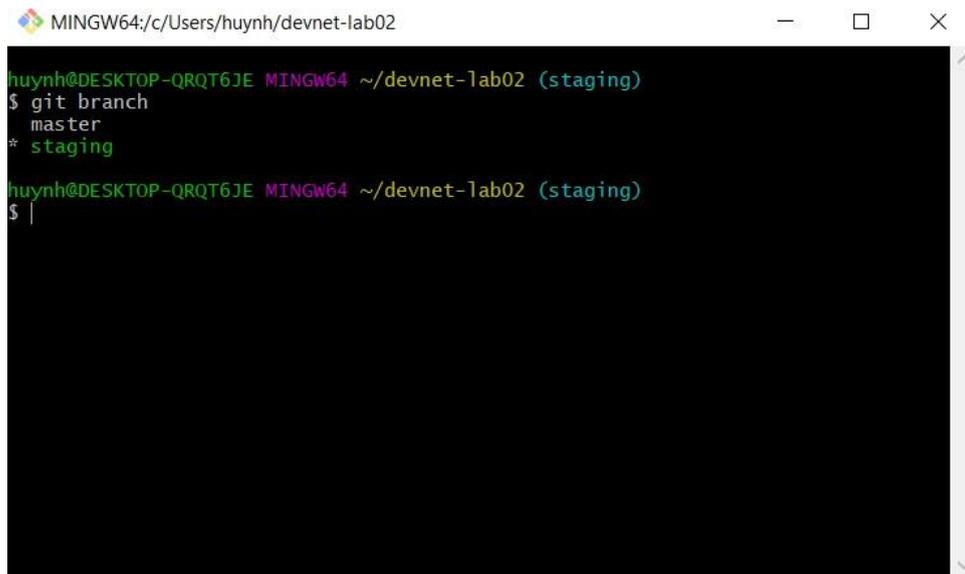
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ git commit -m "Add print statement"
[staging dbd7ddb] Add print statement
1 file changed, 1 insertion(+)

huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ |
```

#### IV. Merge Branch

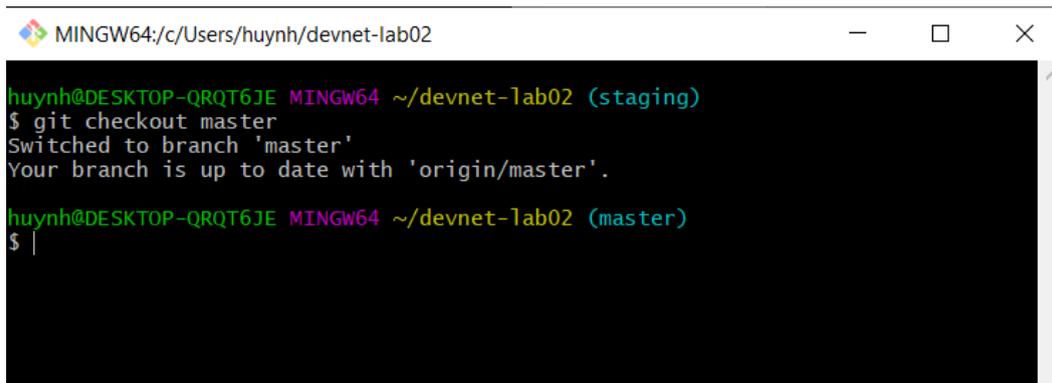
Merge branch là một nhiệm vụ phổ biến khi làm việc với Git. Bạn sẽ liệt kê các nhánh hiện có, kiểm tra xem bạn hiện đang làm việc ở đâu và hợp nhất với nhánh chính mà không xảy ra xung đột.

Để kiểm tra các branch hiện có sử dụng lệnh *git branch*. Ngôi sao phía trước tên sẽ thể hiện nhánh bạn đang làm việc:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ git branch
master
* staging
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ |
```

Trước khi merge nhánh staging với nhánh master, chúng ta phải chuyển về nhánh master bằng câu lệnh *git checkout master*:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (staging)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

Sử dụng câu lệnh *git merge staging* để merge nhánh staging vào nhánh master:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git merge staging
Updating f620738..dbd7ddb
Fast-forward
 lab02.py | 1 +
 1 file changed, 1 insertion(+)
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$
```

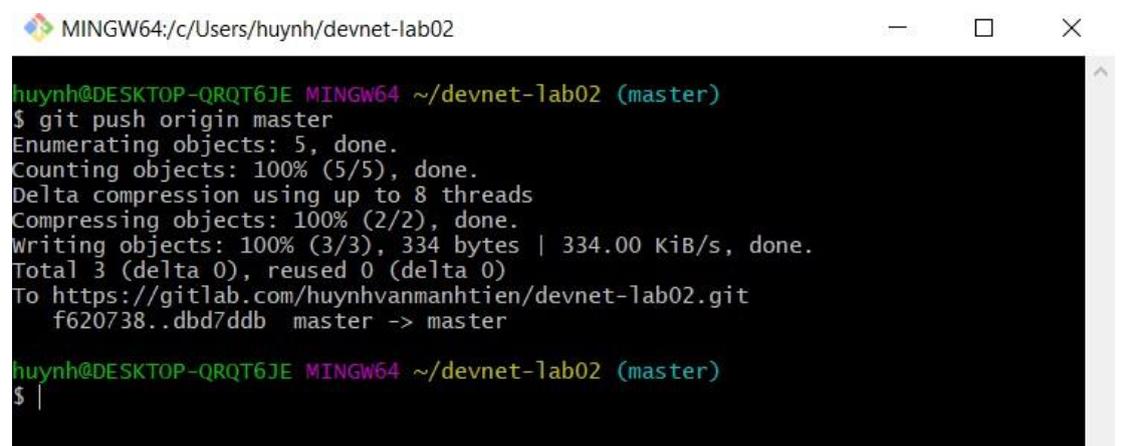
Kiểm tra lại trạng thái của local repository:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

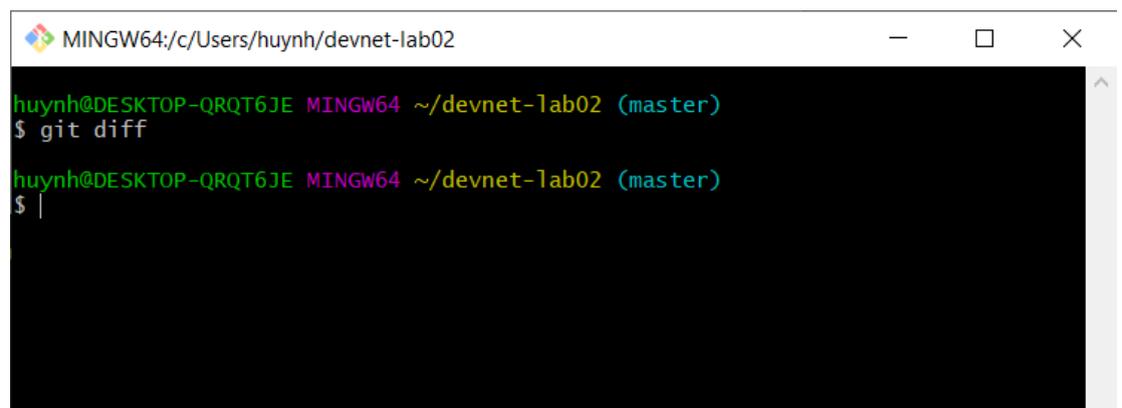
nothing to commit, working tree clean
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

Sau đó push những thay đổi tới remote branch:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 334 bytes | 334.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://gitlab.com/huynhvmanhtien/devnet-lab02.git
   f620738..dbd7ddb master -> master
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```

Lúc này ta nhập lệnh *git diff* sẽ không thấy sự khác nhau giữa nhánh master và nhánh staging nữa:



```
MINGW64:/c/Users/huynh/devnet-lab02
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ git diff
huynh@DESKTOP-QRQT6JE MINGW64 ~/devnet-lab02 (master)
$ |
```