

# Open-Source Programming

## #3: Open-source Management & Github Usage (2-1)



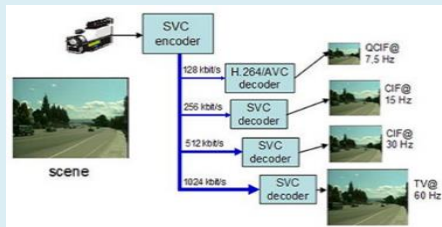
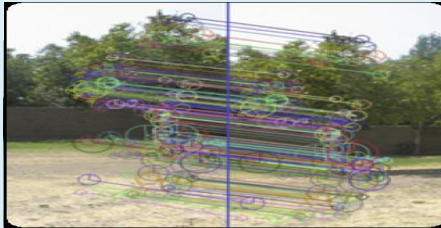
2024 Spring

Prof. Byung-Gyu Kim  
Intelligent Vision Processing Lab. (IVPL)

<http://ivpl.sookmyung.ac.kr>

Dept. of IT Engineering, Sookmyung Women's University

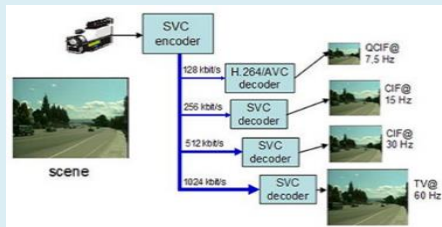
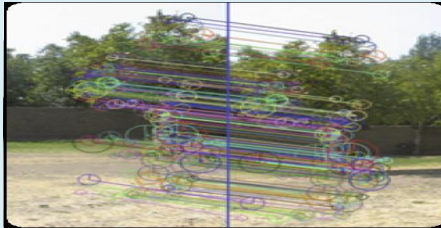
E-mail: [bg.kim@sookmyung.ac.kr](mailto:bg.kim@sookmyung.ac.kr)



## Contents

---

- How to use Git?
- Github + VS Code



## Contents

---

- How to use Git?
- Github + VS Code

# How to use Git? (1)

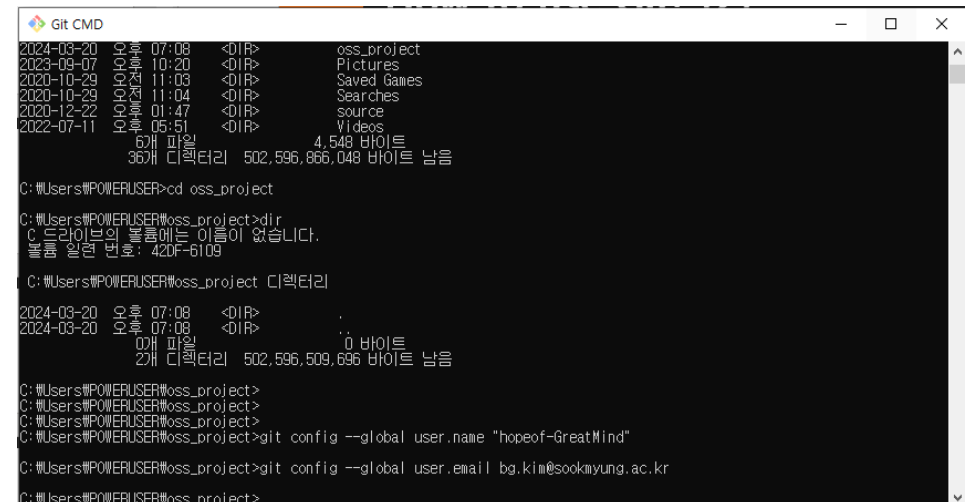
## ❖ Git Installation

- [Git, Git Bash 쉬운 설치/ Git Bash 설치 쉽고 자세한 설명/ 윈도우 OS에서 리눅스 환경 구축 하기/ Git Bash란 무엇인가 \(tistory.com\)](#)
- [Git Guides - install git \(github.com\)](#)

## ❖ Github Account setup and activation

## ❖ Registration for Local User Infor.

```
$ git config --global user.name "Your Name"  
$ git config --global user.email you@example.com
```



```
Git CMD  
2024-03-20 오후 07:08 <DIR> oss_project  
2023-09-07 오후 10:20 <DIR> Pictures  
2020-10-29 오전 11:03 <DIR> Saved Games  
2020-10-29 오전 11:04 <DIR> Searches  
2020-12-22 오후 01:47 <DIR> source  
2022-07-11 오후 05:51 <DIR> Videos  
6개 파일 4,548 바이트  
36개 디렉터리 502,596,866 바이트 남음  
C:\Users\POWERUSER>cd oss_project  
C:\Users\POWERUSER\oss_project>dir  
C 드라이브의 볼륨에는 이름이 없습니다.  
볼륨 일련 번호: 42DF-6109  
C:\Users\POWERUSER\oss_project 디렉터리  
2024-03-20 오후 07:08 <DIR> .  
2024-03-20 오후 07:08 <DIR> ..  
0개 파일 0 바이트  
2개 디렉터리 502,596,509,696 바이트 남음  
C:\Users\POWERUSER\oss_project>  
C:\Users\POWERUSER\oss_project>  
C:\Users\POWERUSER\oss_project>  
C:\Users\POWERUSER\oss_project>git config --global user.name "hopeof-GreatMind"  
C:\Users\POWERUSER\oss_project>git config --global user.email bg.kim@sookmyung.ac.kr  
C:\Users\POWERUSER\oss_project>
```

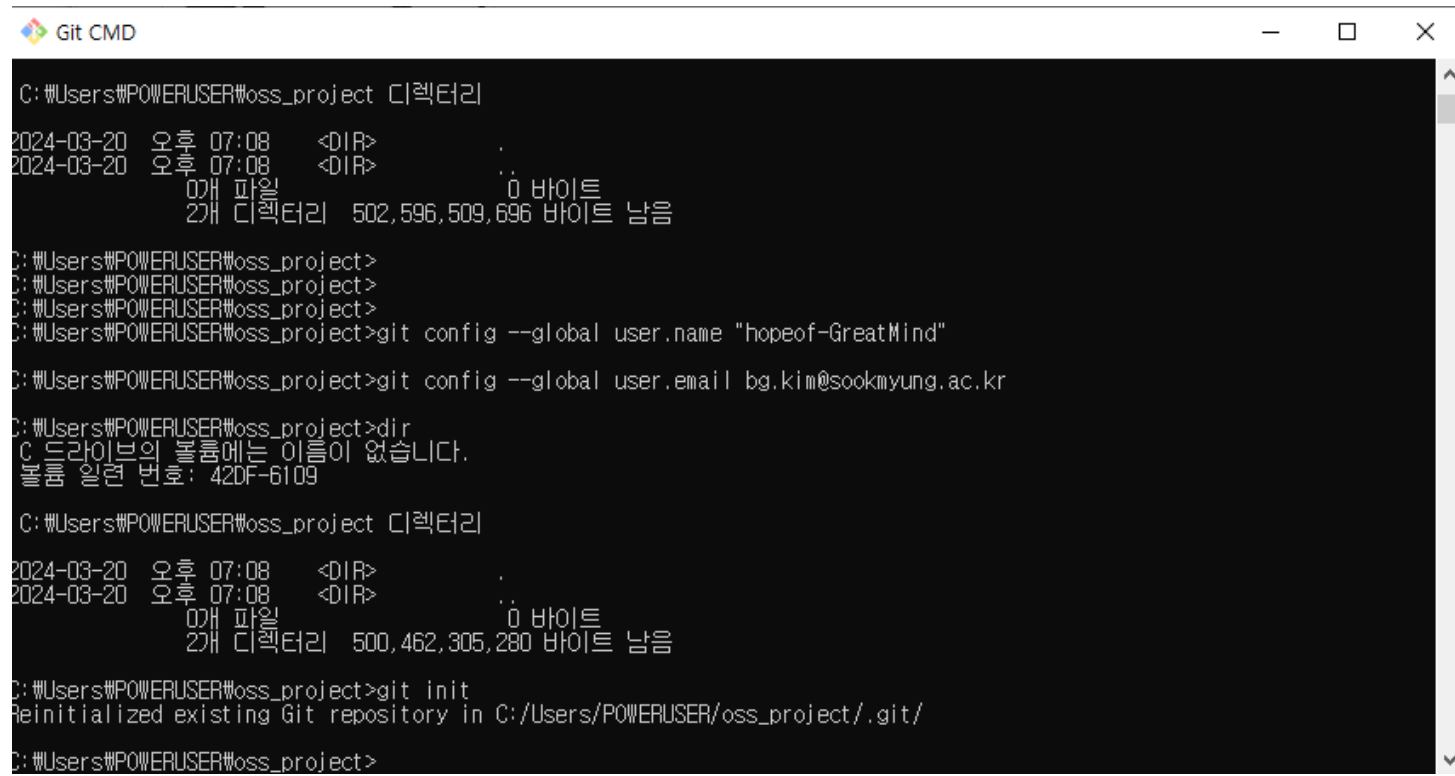
# How to use Git? (2)

## ❖ The process of creating a repository on GitHub and linking it with a local Git repository

- **\$git init** : Initialize your repository if you make the first repository.
- **\$git clone**: Copy your own repository in Github.

- “\$git init” commend:

- 1] First, move to your work folder where you want.
- 2] Type “git init” and click “enter”.



```
Git CMD
C:\Users\POWERUSER\oss_project 디렉터리
2024-03-20 오후 07:08 <DIR>
2024-03-20 오후 07:08 <DIR>
0개 파일 0 바이트
2개 디렉터리 502,596,509,696 바이트 남음

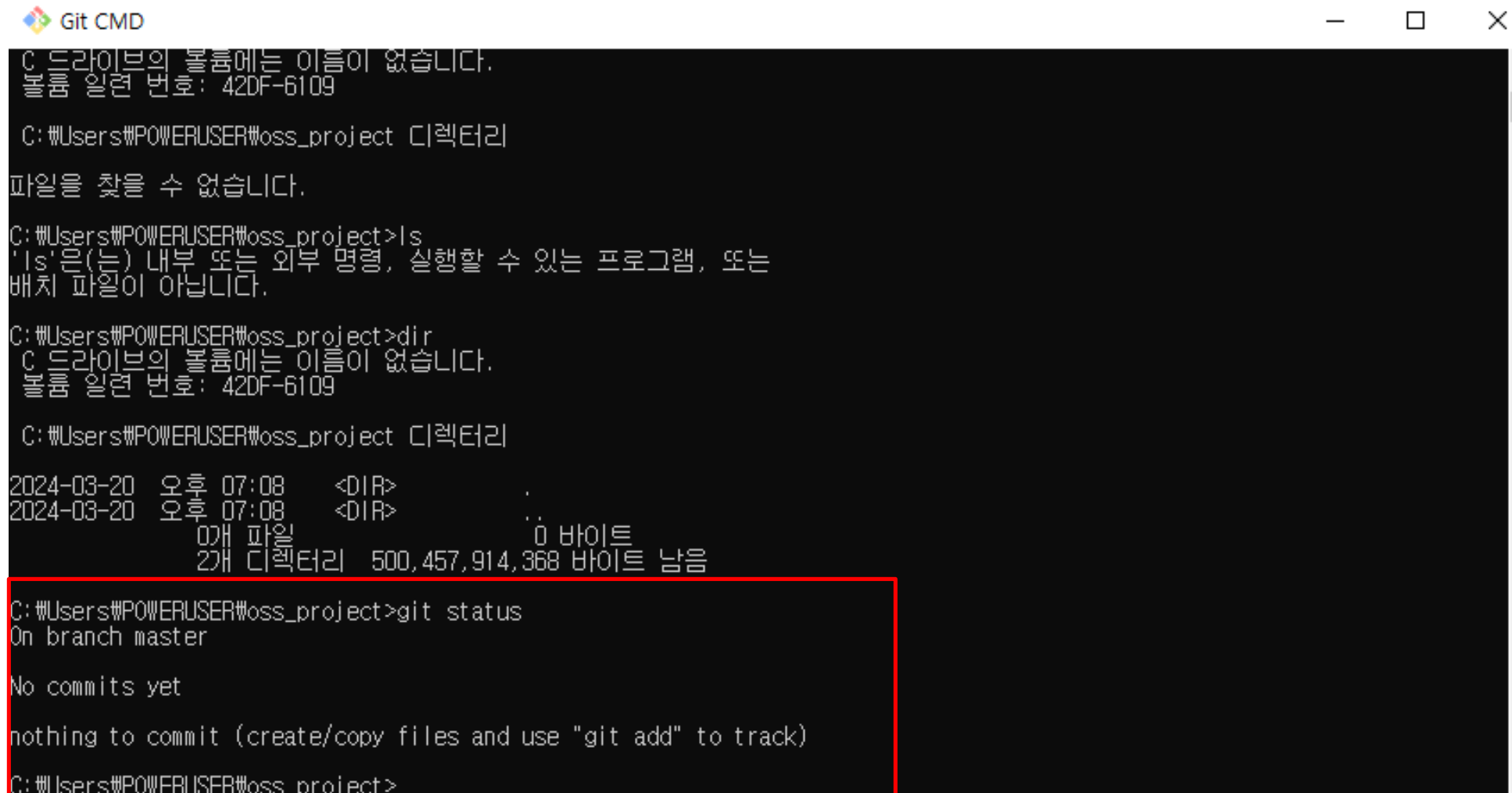
C:\Users\POWERUSER\oss_project>
C:\Users\POWERUSER\oss_project>
C:\Users\POWERUSER\oss_project>
C:\Users\POWERUSER\oss_project>git config --global user.name "hopeof-GreatMind"
C:\Users\POWERUSER\oss_project>git config --global user.email bg.kim@sookmyung.ac.kr
C:\Users\POWERUSER\oss_project>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 42DF-6109

C:\Users\POWERUSER\oss_project 디렉터리
2024-03-20 오후 07:08 <DIR>
2024-03-20 오후 07:08 <DIR>
0개 파일 0 바이트
2개 디렉터리 500,462,305,280 바이트 남음

C:\Users\POWERUSER\oss_project>git init
Reinitialized existing Git repository in C:/Users/POWERUSER/oss_project/.git/
C:\Users\POWERUSER\oss_project>
```

# How to use Git? (3)

- 3] "git status" and check on the message!



```
Git CMD
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 42DF-6109

C:\Users\POWERUSER\oss_project 디렉터리

파일을 찾을 수 없습니다.

C:\Users\POWERUSER\oss_project>ls
'ls'은(는) 내부 또는 외부 명령, 실행할 수 있는 프로그램, 또는
배치 파일이 아닙니다.

C:\Users\POWERUSER\oss_project>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 42DF-6109

C:\Users\POWERUSER\oss_project 디렉터리

2024-03-20 오후 07:08 <DIR> .
2024-03-20 오후 07:08 <DIR> ..
                0개 파일                0 바이트
                2개 디렉터리  500,457,914,368 바이트 남음

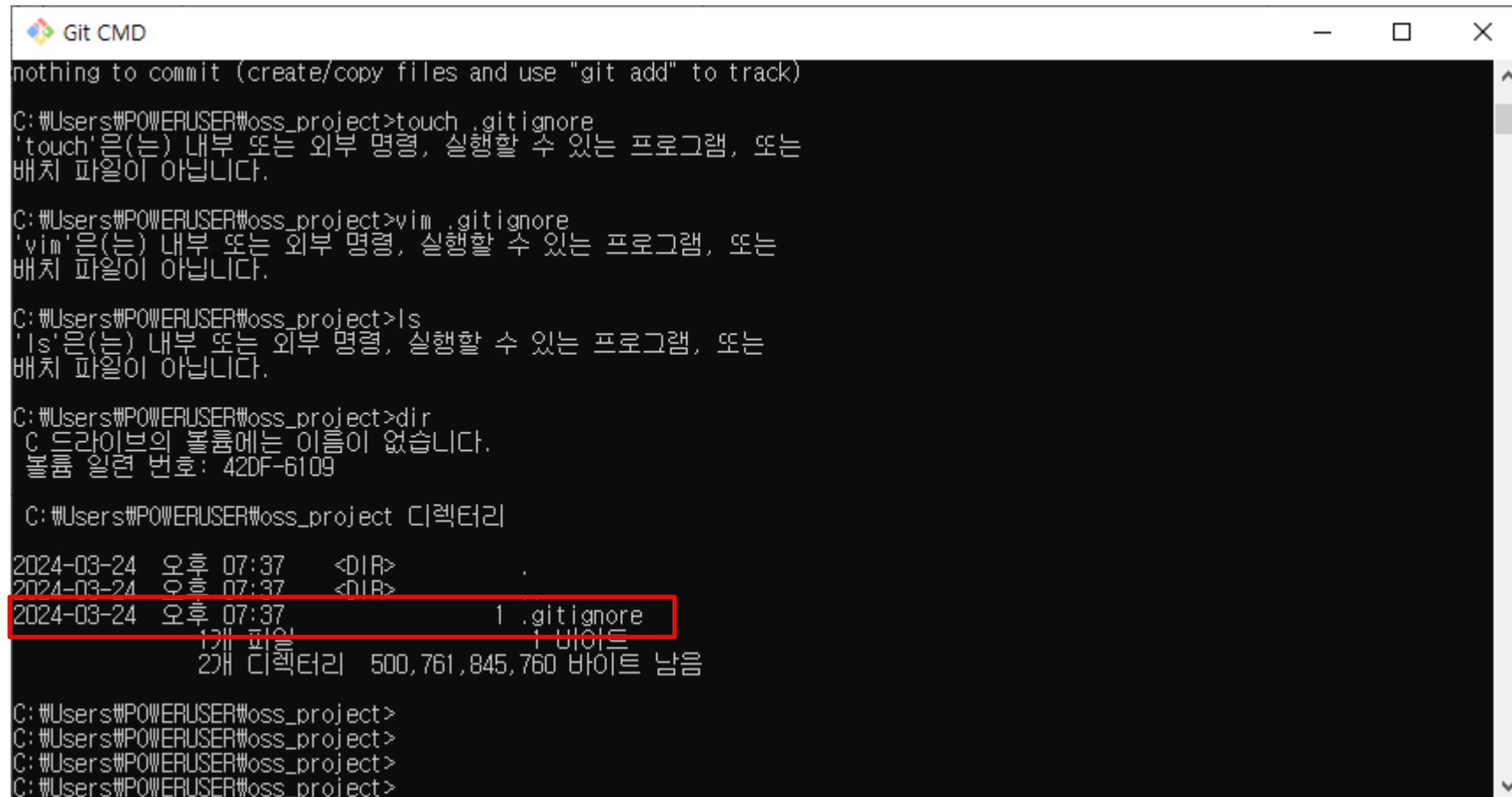
C:\Users\POWERUSER\oss_project>git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
C:\Users\POWERUSER\oss project>
```

## How to use Git? (4)

- 4] Make one dummy file as ".gitignore" in your folder, by using your editor.



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

C:\Users\POWERUSER\oss_project>touch .gitignore
'touch'은(는) 내부 또는 외부 명령, 실행할 수 있는 프로그램, 또는
배치 파일이 아닙니다.

C:\Users\POWERUSER\oss_project>vim .gitignore
'vim'은(는) 내부 또는 외부 명령, 실행할 수 있는 프로그램, 또는
배치 파일이 아닙니다.

C:\Users\POWERUSER\oss_project>ls
'ls'은(는) 내부 또는 외부 명령, 실행할 수 있는 프로그램, 또는
배치 파일이 아닙니다.

C:\Users\POWERUSER\oss_project>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 42DF-6109

C:\Users\POWERUSER\oss_project 디렉터리

2024-03-24 오후 07:37 <DIR> .
2024-03-24 오후 07:37 <DIR> ..
2024-03-24 오후 07:37 1 .gitignore
1개 파일 1 바이트
2개 디렉터리 500,761,845,760 바이트 남음

C:\Users\POWERUSER\oss_project>
C:\Users\POWERUSER\oss_project>
C:\Users\POWERUSER\oss_project>
C:\Users\POWERUSER\oss_project>
```

# How to use Git? (5)

- 5] Configure user name and **Primary e-mail** as:

```
$ git config --global user.email "you@example.com"
$ git config --global user.name "Your Name"
```

- 6 Execute the following commands:

```
$ git add .gitignore
$ git commit -m "by using git init method"
```

- 7] Move to the "main" branch:

```
$ git branch -M main (예전: master)
```

```
$git branch    <-- 현재 브랜치 확인
```

- 8] 수정 기록 확인

```
$ git log
```

```
POWERUSER@DESKTOP-PEEULI9 MINGW64 /tmp/literate-engine (main)
$ git log
commit 0826fc8146924f78a375be99aa1639dd875de467 (HEAD -> main)
Author: GreatMind <bg.kim@sookmyung.ac.kr>
Date: Tue Apr 9 19:36:36 2024 +0900

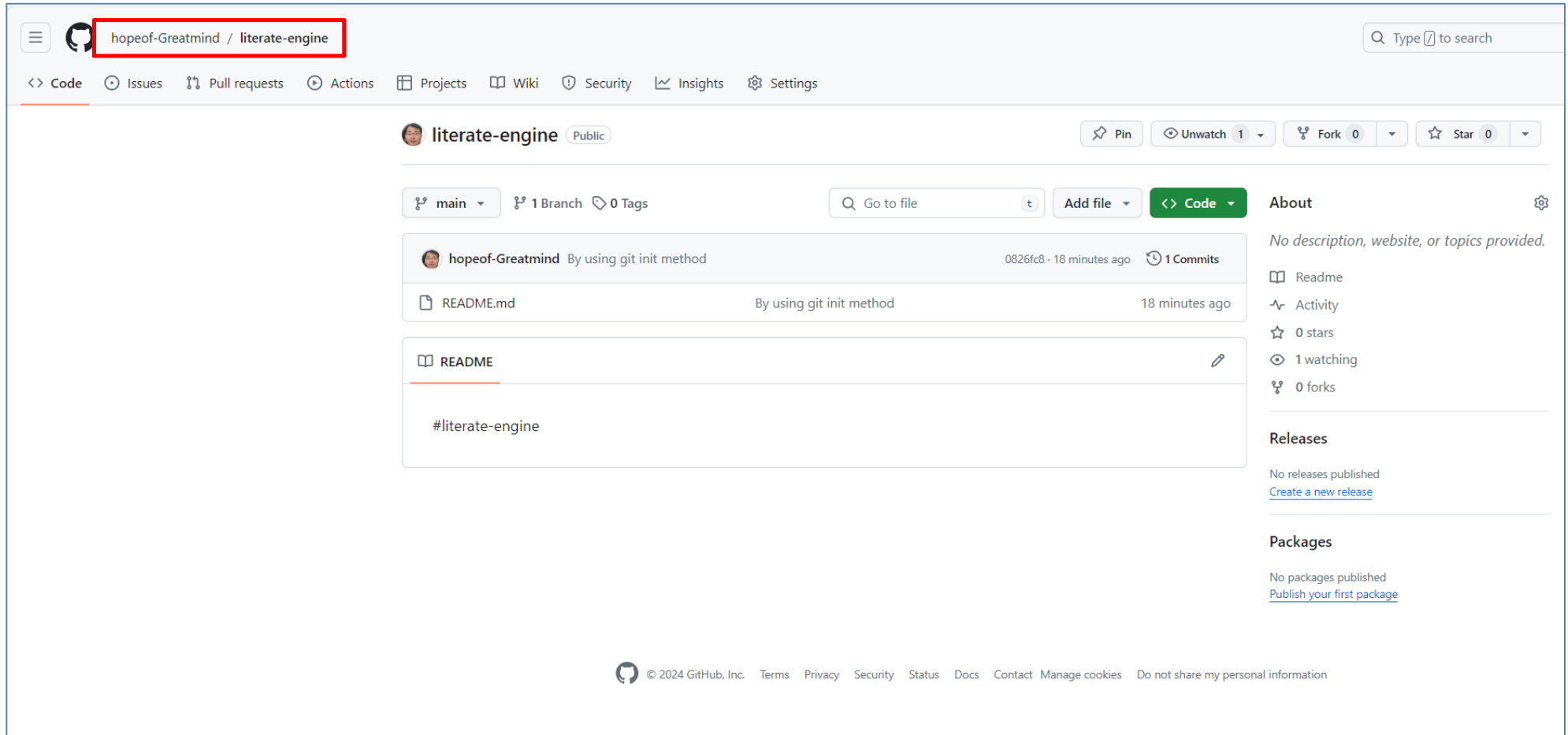
    By using git init method

POWERUSER@DESKTOP-PEEULI9 MINGW64 /tmp/literate-engine (main)
```



# How to use Git? (6)

- 9] Make your blank repository in Github site: Here, we use "literate-engine".



# How to use Git? (7)

- 10] Connect your local repository to your Github repository as:

```
$ git remote add origin https://github.com/hopeof-Greatmind/literate-engine.git
```

- 11] Verify the remote repository as:

```
$ git remote -v
```

- 12] Push your local source to your Github as:

```
$ git push -u origin main (예전: master)
```

```
POWERUSER@DESKTOP-PEEULI9 MINGW64 /tmp/literate-engine (main)
$ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 245 bytes | 245.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/hopeof-Greatmind/literate-engine.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

POWERUSER@DESKTOP-PEEULI9 MINGW64 /tmp/literate-engine (main)
```

# How to use Git? (8)

- 12] Check on your Github site for uploading your local sources....!!!

The screenshot shows the GitHub interface for the repository 'hopeof-Greatmind / literate-engine'. The repository is public and has 1 commit. A red dashed box highlights the README file, which contains the text '#literate-engine'. A red arrow points from the text 'Committed file...!!!' to the README file.

**Committed file...!!!**

# How to use Git? (6)

## ■ “\$git clone” Command

```
git clone [REPO_URL] [DIR]
```

- 1] Check your Github URL in Github!

# HTTPS 형식

`https://github.com/[USERNAME]/[reps_name].git`

# SSH 형식

`git@github.com:[username]/[reps_name].git`

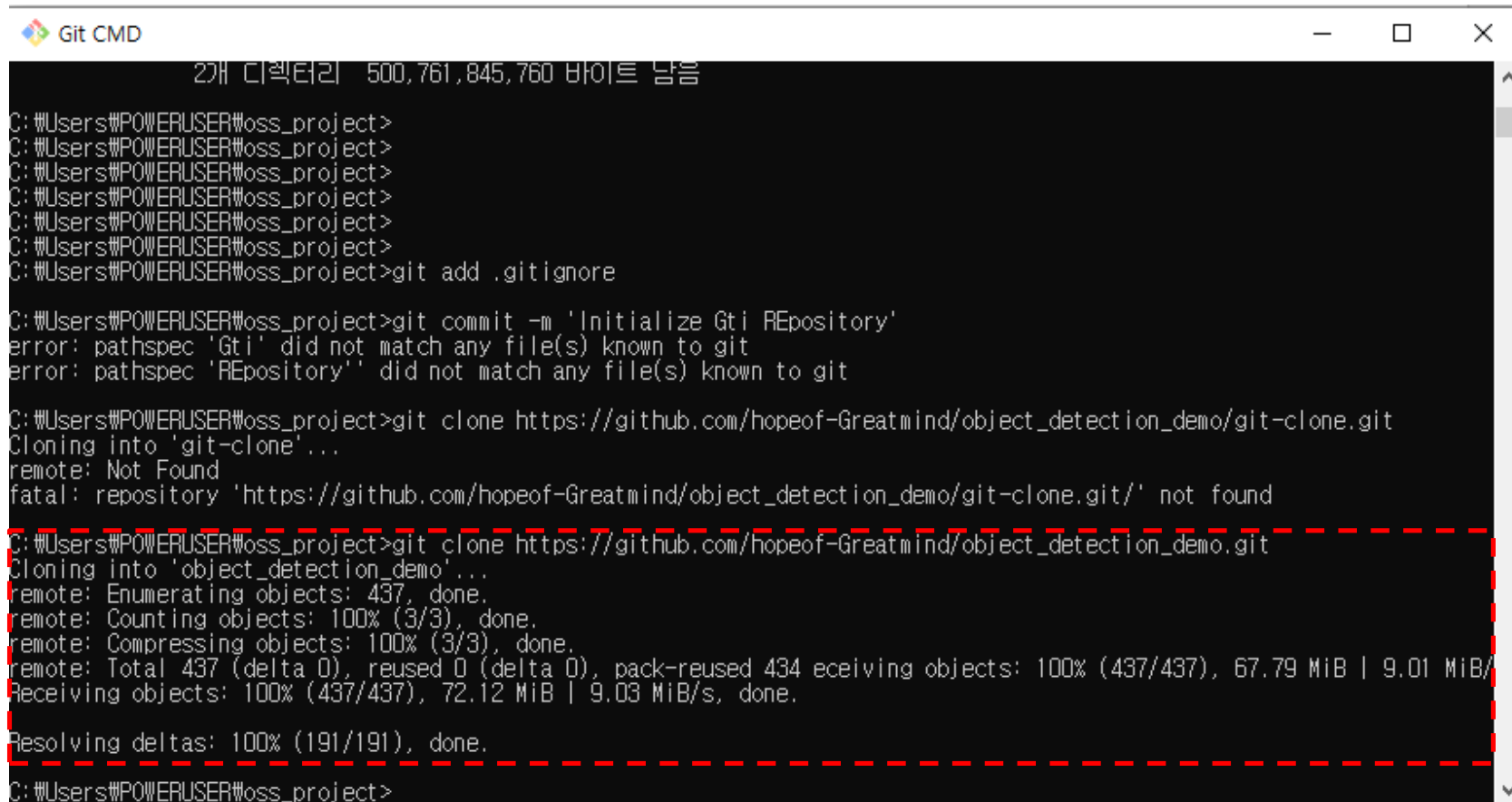
The screenshot displays the GitHub interface for the repository 'object\_detection\_demo' by user 'hopeof-Greatmind'. The repository is public and has 28 commits. The file list shows various files and folders, including 'data', 'deploy', 'test', 'export\_inference\_graph.py', 'generate\_tfrecord.py', 'local\_inference\_test.ipynb', 'local\_inference\_test.py', 'requirements.txt', 'resize\_images.py', 'tensorflow\_object\_detection\_training\_colab.ipynb', 'tensorflow\_object\_detection\_training\_colab.ipynb', and 'xml\_to\_csv.py'. The right sidebar shows repository statistics: 0 stars, 1 watching, 0 forks. The bottom section shows suggested workflows, including 'Publish Python Package'.

# How to use Git? (7)

- 2] Copy the source using the following command:

```
$ git clone https://github.com/hopeof-Greatmind/object_detection_demo.git
```

You can see cloning process as the following:



```
Git CMD
2개 디렉터리 500,761,845,760 바이트 남음

C:\Users\#POWERUSER\#oss_project>
C:\Users\#POWERUSER\#oss_project>
C:\Users\#POWERUSER\#oss_project>
C:\Users\#POWERUSER\#oss_project>
C:\Users\#POWERUSER\#oss_project>
C:\Users\#POWERUSER\#oss_project>
C:\Users\#POWERUSER\#oss_project>git add .gitignore

C:\Users\#POWERUSER\#oss_project>git commit -m 'Initialize Gti REpository'
error: pathspec 'Gti' did not match any file(s) known to git
error: pathspec 'REpository' did not match any file(s) known to git

C:\Users\#POWERUSER\#oss_project>git clone https://github.com/hopeof-Greatmind/object_detection_demo/git-clone.git
Cloning into 'git-clone'...
remote: Not Found
fatal: repository 'https://github.com/hopeof-Greatmind/object_detection_demo/git-clone.git/' not found

C:\Users\#POWERUSER\#oss_project>git clone https://github.com/hopeof-Greatmind/object_detection_demo.git
Cloning into 'object_detection_demo'...
remote: Enumerating objects: 437, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 437 (delta 0), reused 0 (delta 0), pack-reused 434 eceiving objects: 100% (437/437), 67.79 MiB | 9.01 MiB/s
Receiving objects: 100% (437/437), 72.12 MiB | 9.03 MiB/s, done.
Resolving deltas: 100% (191/191), done.

C:\Users\#POWERUSER\#oss_project>
```

# How to use Git? (8)

- 3] Check on your folder to verify the cloning!

```
Git CMD
C:\Users\POWERUSER\oss_project>git clone https://github.com/hopeof-Greatmind/object_detection_demo/git-clone.git
Cloning into 'git-clone'...
remote: Not Found
fatal: repository 'https://github.com/hopeof-Greatmind/object_detection_demo/git-clone.git/' not found

C:\Users\POWERUSER\oss_project>git clone https://github.com/hopeof-Greatmind/object_detection_demo.git
Cloning into 'object_detection_demo'...
remote: Enumerating objects: 437, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 437 (delta 0), reused 0 (delta 0), pack-reused 434
Receiving objects: 100% (437/437), 67.79 MiB | 9.01 MiB/s, done.
Resolving deltas: 100% (191/191), done.

C:\Users\POWERUSER\oss_project>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 42DF-6109

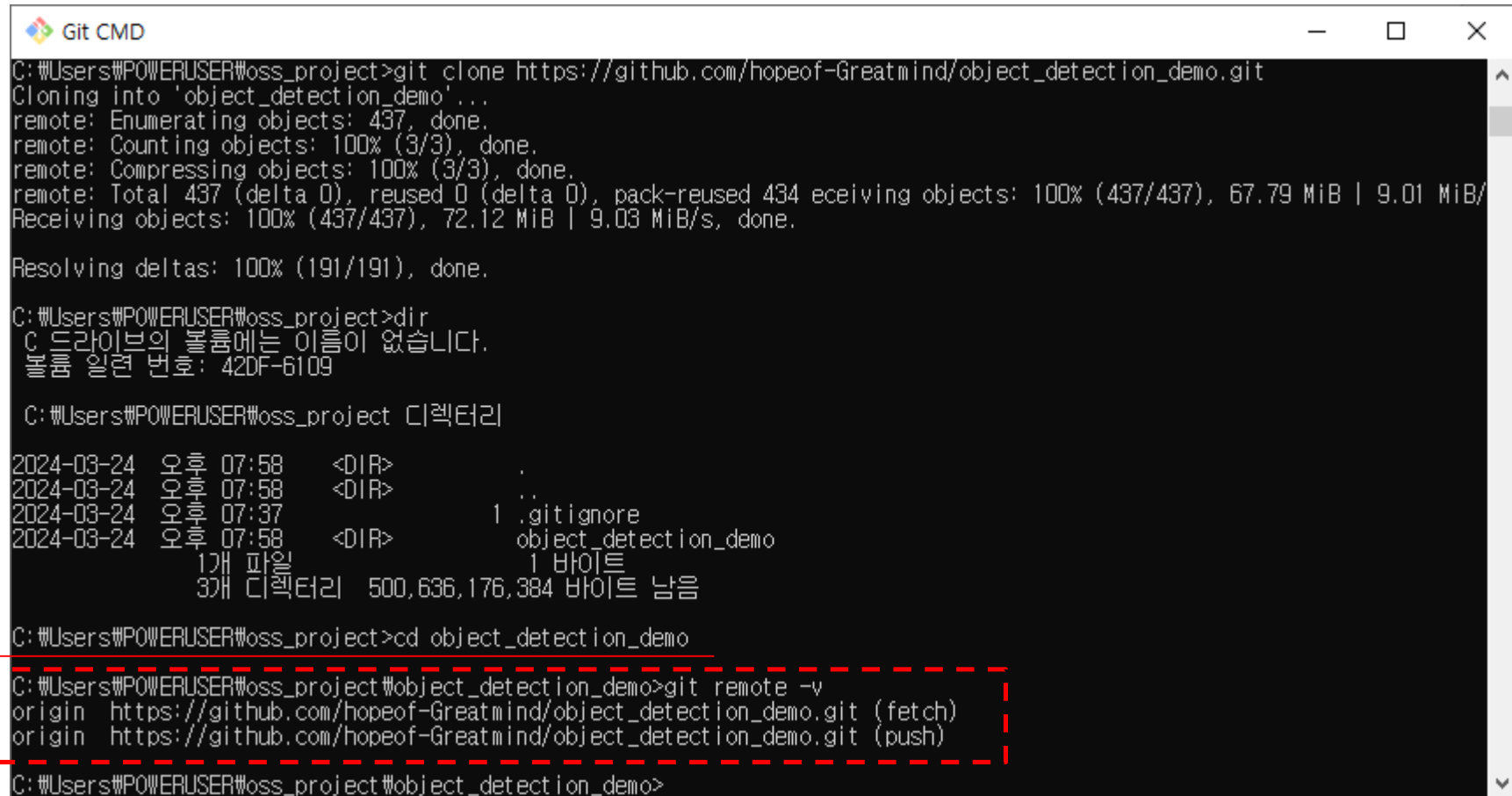
C:\Users\POWERUSER\oss_project 디렉터리
2024-03-24 오후 07:58 <DIR> .
2024-03-24 오후 07:58 <DIR> ..
2024-03-24 오후 07:37 -gitignore-
2024-03-24 오후 07:58 <DIR> object_detection_demo
1개 파일 1 바이트
3개 디렉터리 500,636,176,384 바이트 남음

C:\Users\POWERUSER\oss_project>
```

# How to use Git? (9)

- 4] Verify the remote repository.

```
$ cd [your clone folder]
$ git remote -v
```



```
Git CMD
C:\Users\POWERUSER\oss_project>git clone https://github.com/hopeof-Greatmind/object_detection_demo.git
Cloning into 'object_detection_demo'...
remote: Enumerating objects: 437, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 437 (delta 0), reused 0 (delta 0), pack-reused 434 receiving objects: 100% (437/437), 67.79 MiB | 9.01 MiB/s
Receiving objects: 100% (437/437), 72.12 MiB | 9.03 MiB/s, done.
Resolving deltas: 100% (191/191), done.
C:\Users\POWERUSER\oss_project>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 42DF-6109

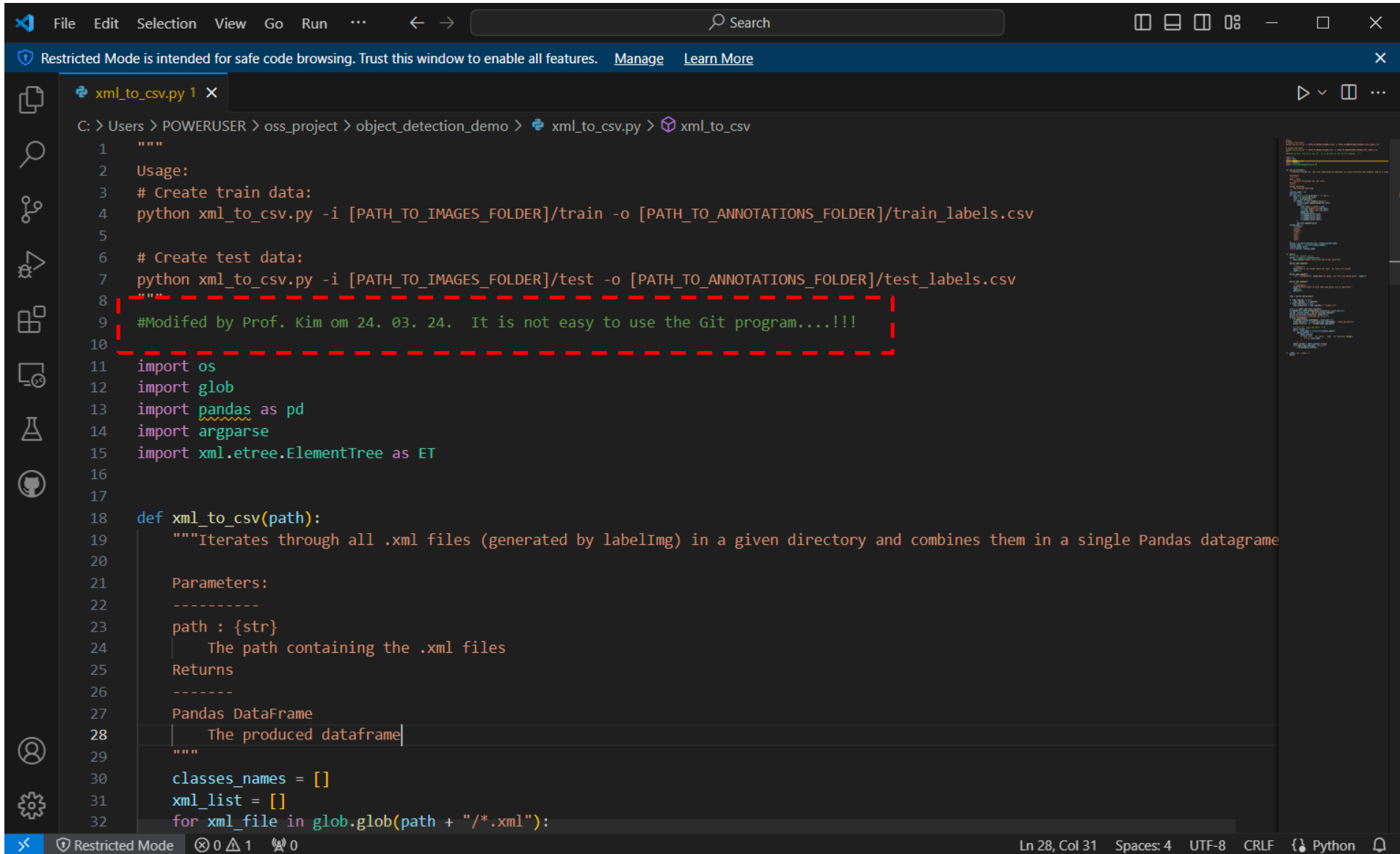
C:\Users\POWERUSER\oss_project 디렉터리

2024-03-24 오후 07:58 <DIR>          .
2024-03-24 오후 07:58 <DIR>          ..
2024-03-24 오후 07:37          1 .gitignore
2024-03-24 오후 07:58 <DIR>          object_detection_demo
                1개 파일              1 바이트
                3개 디렉터리  500,636,176,384 바이트 남음

C:\Users\POWERUSER\oss_project>cd object_detection_demo
C:\Users\POWERUSER\oss_project\object_detection_demo>git remote -v
origin  https://github.com/hopeof-Greatmind/object_detection_demo.git (fetch)
origin  https://github.com/hopeof-Greatmind/object_detection_demo.git (push)
C:\Users\POWERUSER\oss_project\object_detection_demo>
```

# How to use Git? (10)

- 5] Edit your code and modification. Here, we use "xml\_2\_csv.py" file.



The screenshot shows a code editor window with the file `xml_to_csv.py` open. The editor has a menu bar (File, Edit, Selection, View, Go, Run, ...) and a search bar. A notification bar at the top states: "Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More". The file path in the editor is `C:\> Users > POWERUSER > oss_project > object_detection_demo > xml_to_csv.py > xml_to_csv`. The code content is as follows:

```
1  """
2  Usage:
3  # Create train data:
4  python xml_to_csv.py -i [PATH_TO_IMAGES_FOLDER]/train -o [PATH_TO_ANNOTATIONS_FOLDER]/train_labels.csv
5
6  # Create test data:
7  python xml_to_csv.py -i [PATH_TO_IMAGES_FOLDER]/test -o [PATH_TO_ANNOTATIONS_FOLDER]/test_labels.csv
8  """
9  #Modified by Prof. Kim om 24. 03. 24. It is not easy to use the Git program....!!!
10
11 import os
12 import glob
13 import pandas as pd
14 import argparse
15 import xml.etree.ElementTree as ET
16
17
18 def xml_to_csv(path):
19     """Iterates through all .xml files (generated by labelImg) in a given directory and combines them in a single Pandas dataframe
20
21     Parameters:
22     -----
23     path : {str}
24         The path containing the .xml files
25     Returns
26     -----
27     Pandas DataFrame
28         The produced dataframe
29     """
30     classes_names = []
31     xml_list = []
32     for xml_file in glob.glob(path + "/*.xml"):
```

The modification on line 9 is highlighted with a red dashed box. The status bar at the bottom shows "Ln 28, Col 31", "Spaces: 4", "UTF-8", "CRLF", "Python", and a bell icon.



# How to use Git? (11)

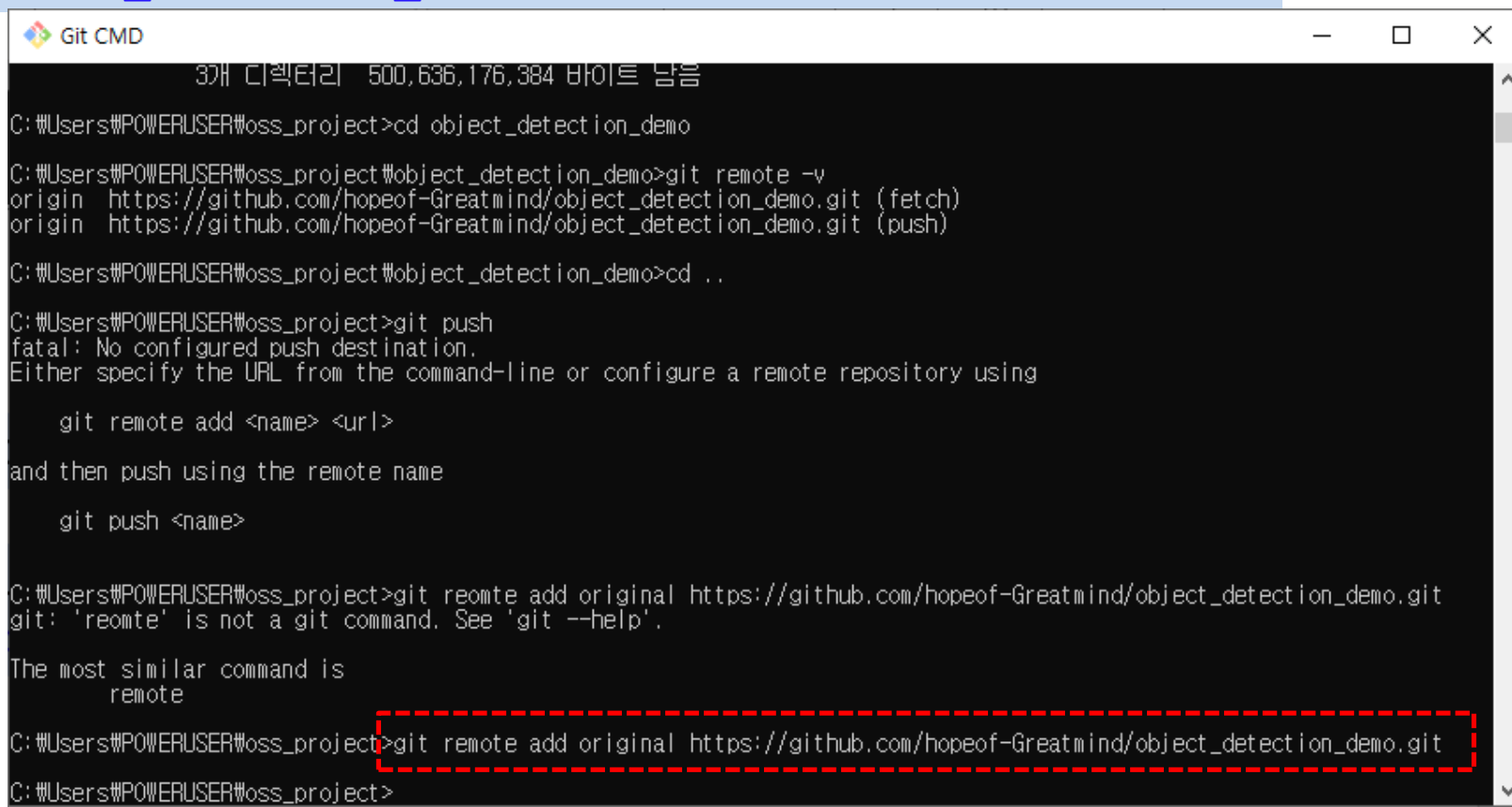
- 6] Before reflecting the change to the remote repository, **connect two repositories** based on "git remote" command.

원격 저장소의 이름으로 사용할 문자열

```
$ git remote add [REMOTE_NAME] [REMOTE_GIT_URL]
```

원격 Git 저장소 주소를 지정

```
$ git remote add origin https://github.com/hopeof-Greatmind/object_detection_demo.git
```



```
Git CMD
3개 디렉터리 500,636,176 바이트 남음
C:\Users\POWERUSER\oss_project>cd object_detection_demo
C:\Users\POWERUSER\oss_project\object_detection_demo>git remote -v
origin https://github.com/hopeof-Greatmind/object_detection_demo.git (fetch)
origin https://github.com/hopeof-Greatmind/object_detection_demo.git (push)
C:\Users\POWERUSER\oss_project\object_detection_demo>cd ..
C:\Users\POWERUSER\oss_project>git push
fatal: No configured push destination.
Either specify the URL from the command-line or configure a remote repository using

    git remote add <name> <url>

and then push using the remote name

    git push <name>

C:\Users\POWERUSER\oss_project>git reomte add original https://github.com/hopeof-Greatmind/object_detection_demo.git
git: 'reomte' is not a git command. See 'git --help'.

The most similar command is
    remote
C:\Users\POWERUSER\oss_project>git remote add original https://github.com/hopeof-Greatmind/object_detection_demo.git
C:\Users\POWERUSER\oss_project>
```

# How to use Git? (12)

- 6] Update git program when you meet an error...!!!!

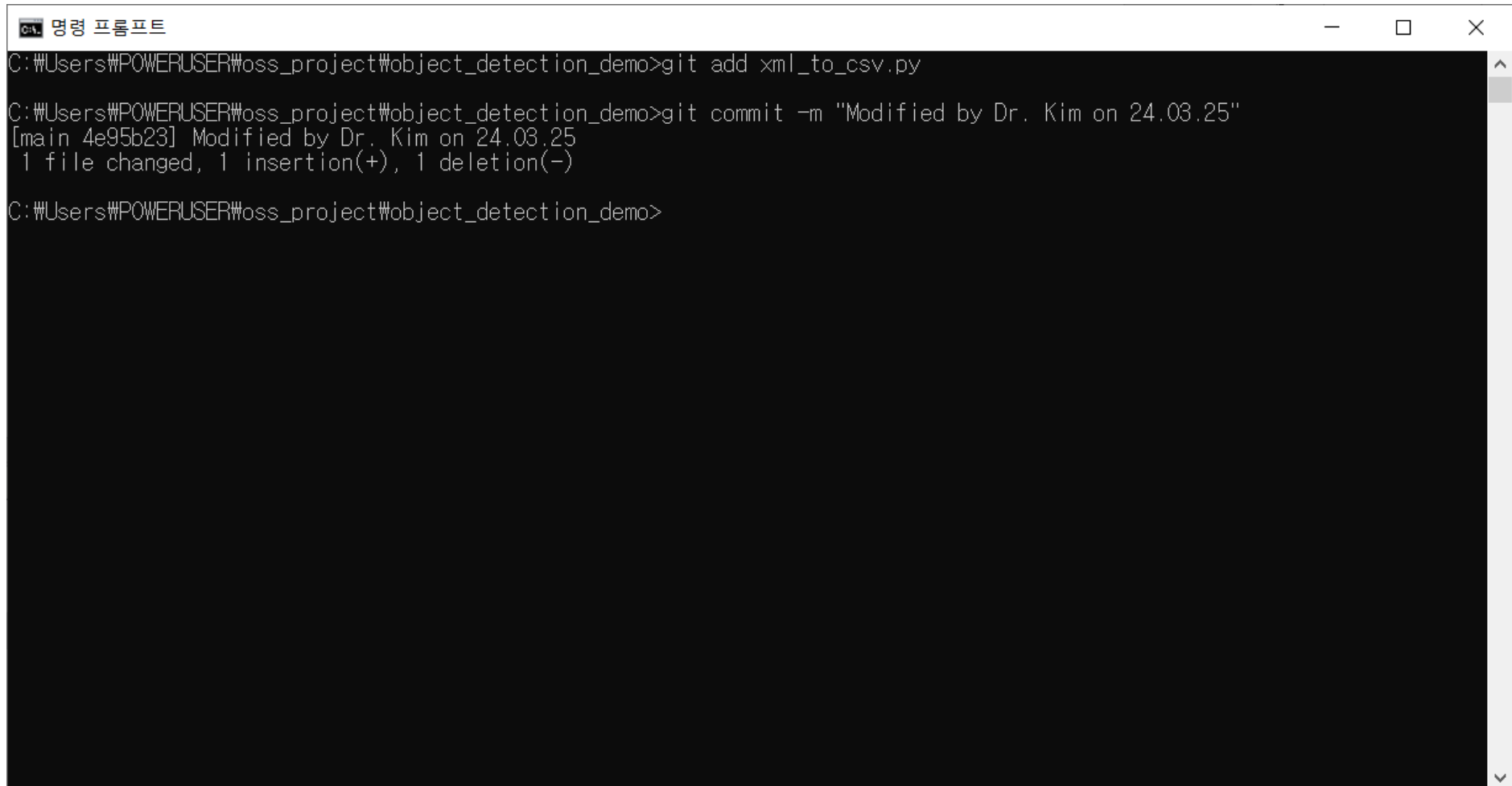
```
$ git update-git-for-windows
```

```
C:\Users\POWERUSER\oss_project\object_detection_demo>git update-git-for-windows
Git for Windows 2.30.0.windows.2 (64bit)
Update 2.44.0.windows.1 is available
Download and install Git for Windows v2.44.0.windows.1 [N/y]? y
##### 100.0%
```

# How to use Git? (13)

- 7] Reflect your change based on "git add <edited file>" and "git Commit -m <message>".

```
$ git add xml_2_csv.py  
$ git commit -m "Modified by Dr. Kim on 24.03.25"
```



A screenshot of a Windows command prompt window titled "명령 프롬프트" (Command Prompt). The window shows the execution of two Git commands. The first command is "git add xml\_to\_csv.py", and the second is "git commit -m 'Modified by Dr. Kim on 24.03.25'". The output of the second command shows the commit hash "4e95b23", the commit message "Modified by Dr. Kim on 24.03.25", and a summary of changes: "1 file changed, 1 insertion(+), 1 deletion(-)".

```
C:\Users\POWERUSER\oss_project\object_detection_demo>git add xml_to_csv.py  
C:\Users\POWERUSER\oss_project\object_detection_demo>git commit -m "Modified by Dr. Kim on 24.03.25"  
[main 4e95b23] Modified by Dr. Kim on 24.03.25  
1 file changed, 1 insertion(+), 1 deletion(-)  
C:\Users\POWERUSER\oss_project\object_detection_demo>
```

# How to use Git? (14)

- 8] Reflect the change to the remote Github (your Github) using "**git push~**" command...!!

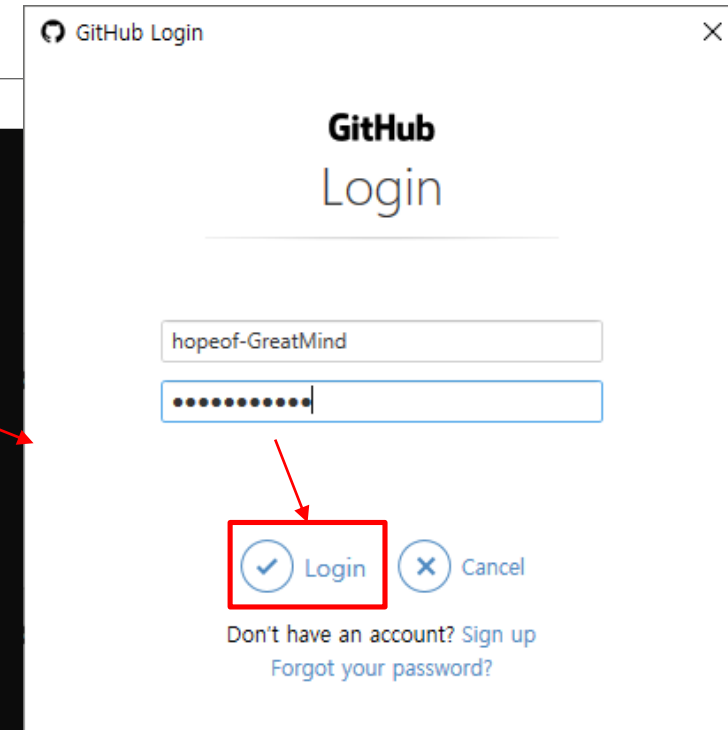
```
$ git push <remote> <branch name>
```

```
Example)) $ git push -u origin main
```

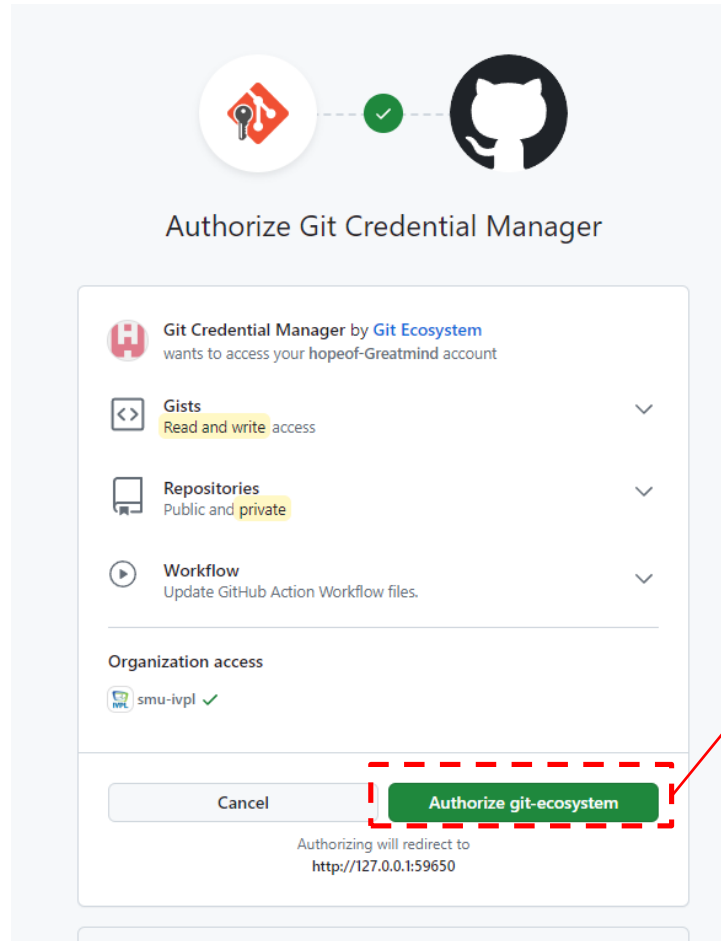
```
$ git push <options>
```

```
명령 프롬프트
C:\Users\POWERUSER\oss_project\object_detection_demo>git add xml_to_csv.py
C:\Users\POWERUSER\oss_project\object_detection_demo>git commit -m "Modified by Dr. Kim on 24.03.25"
[main 4e95b23] Modified by Dr. Kim on 24.03.25
1 file changed, 1 insertion(+), 1 deletion(-)
C:\Users\POWERUSER\oss_project\object_detection_demo>git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 20 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 358 bytes | 358.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/hopeof-GreatMind/object_detection_demo.git
cf119c8..4e95b23 main -> main
branch 'main' set up to track 'origin/main'.
C:\Users\POWERUSER\oss_project\object_detection_demo>
```

*In your working  
Folder !*

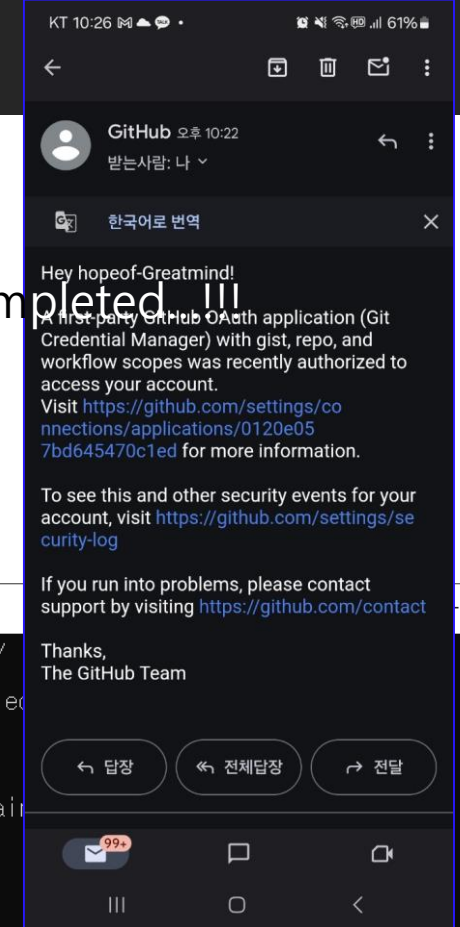


# How to use Git? (15)



Authorization steps completed.!!!

```
C:\Users\hopeof-Greatmind>git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 20 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 358 bytes | 358.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com:hopeof-Greatmind/object_detection_demo.git
   cf119c8..4e95b23  main -> main
branch 'main' set up to track 'origin/main'.
C:\Users\hopeof-Greatmind>
```



# How to use Git? (16)

- 9] See the changes in your remote Github...!!!

hopeof-Greatmind / object\_detection\_demo

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

object\_detection\_demo Public

main had recent pushes 4 minutes ago

Compare & pull request

Switch branches/tags

Find or create a branch...

Branches Tags

✓ master default

main

View all branches

985f877 · 5 years ago 28 Commits

Delete IMG_2383.JPG	5 years ago
Add files via upload	5 years ago
Add files via upload	5 years ago
Add files via upload	5 years ago
Add files via upload	5 years ago
Add files via upload	5 years ago
local_inference_test.ipynb	5 years ago
local_inference_test.py	5 years ago
requirements.txt	5 years ago
resize_images.py	5 years ago
tensorflow_object_detection_training_colab.ipynb	5 years ago
tensorflow_object_detection_training_colab.ipynb	5 years ago
tensorflow_object_detection_training_colab.ipynb	Colaboratory를 통해 생성됨
xml_to_csv.py	5 years ago

README

https://github.com/hopeof-Greatmind/object\_detection\_demo/tree/main

main had recent pushes 6 minutes ago

Compare & pull request

main 2 Branches 0 Tags

Go to file

Add file

Code

This branch is 3 commits ahead of master.

Contribute

hopeof-Greatmind Modified by Dr. Kim on 24.03.25

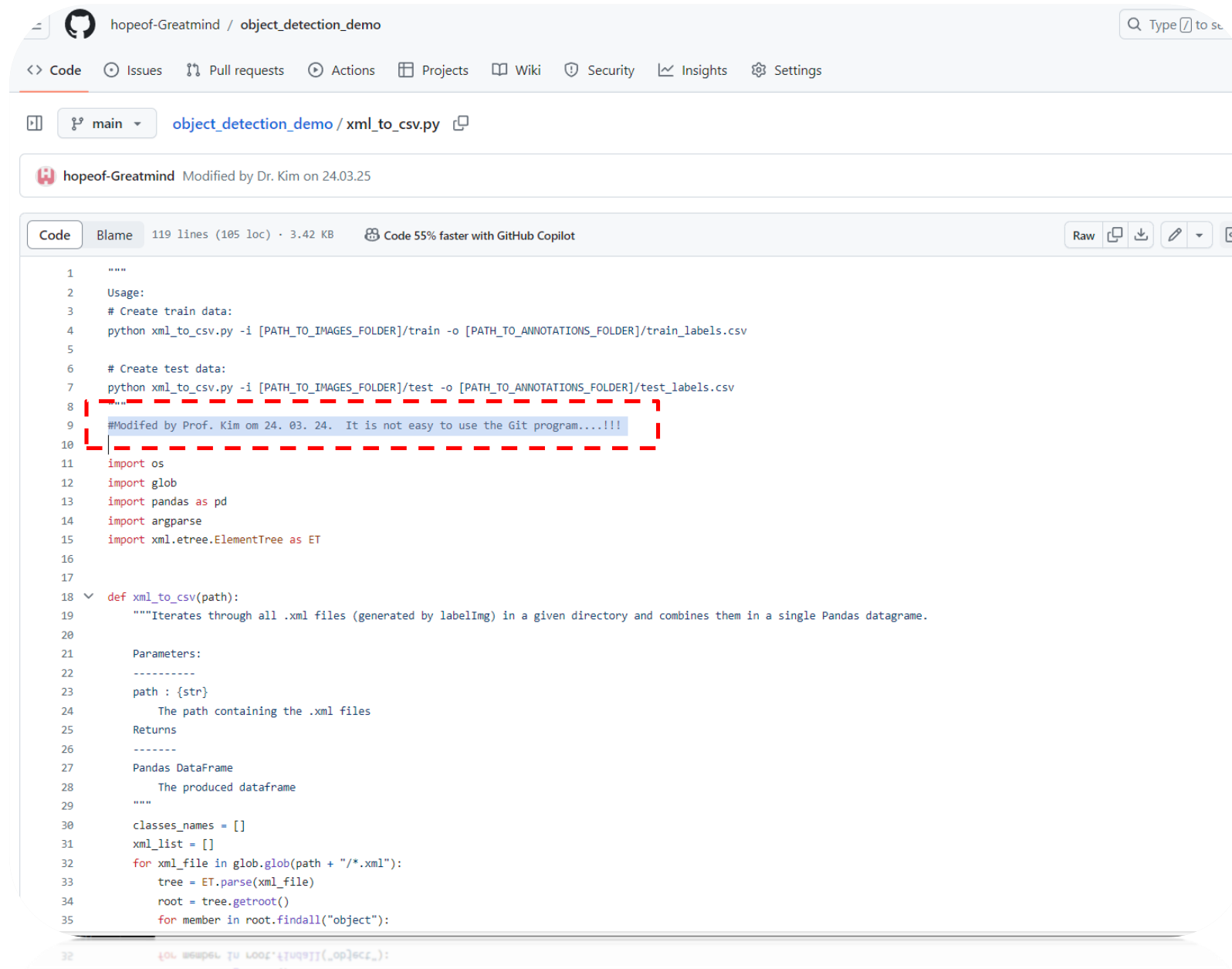
4e95b23 · 8 minutes ago 31 Commits

data	Delete IMG_2383.JPG	5 years ago
deploy	Add files via upload	5 years ago
test	Add files via upload	5 years ago
Readme.md	description of readme file	20 minutes ago
export_inference_graph.py	Add files via upload	5 years ago
generate_tfrecord.py	Add files via upload	5 years ago
local_inference_test.ipynb	Add files via upload	5 years ago
local_inference_test.py	Add files via upload	5 years ago
requirements.txt	Add files via upload	5 years ago
resize_images.py	Add files via upload	5 years ago
tensorflow_object_detection_training_colab.ipynb	Add files via upload	5 years ago
tensorflow_object_detection_training_colab.ipynb	Colaboratory를 통해 생성됨	5 years ago
xml_to_csv.py	Modified by Dr. Kim on 24.03.25	8 minutes ago

README

This is test file for OSS class in Sookmyung Women's University.

# How to use Git? (17)



hopeof-Greatmind / object\_detection\_demo

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main object\_detection\_demo / xml\_to\_csv.py

hopeof-Greatmind Modified by Dr. Kim on 24.03.25

Code Blame 119 lines (105 loc) · 3.42 KB Code 55% faster with GitHub Copilot

```
1  """
2  Usage:
3  # Create train data:
4  python xml_to_csv.py -i [PATH_TO_IMAGES_FOLDER]/train -o [PATH_TO_ANNOTATIONS_FOLDER]/train_labels.csv
5
6  # Create test data:
7  python xml_to_csv.py -i [PATH_TO_IMAGES_FOLDER]/test -o [PATH_TO_ANNOTATIONS_FOLDER]/test_labels.csv
8  """
9  #Modified by Prof. Kim om 24. 03. 24. It is not easy to use the Git program....!!!
10
11 import os
12 import glob
13 import pandas as pd
14 import argparse
15 import xml.etree.ElementTree as ET
16
17
18 def xml_to_csv(path):
19     """Iterates through all .xml files (generated by labeling) in a given directory and combines them in a single Pandas datagramme.
20
21     Parameters:
22     -----
23     path : {str}
24         The path containing the .xml files
25     Returns
26     -----
27     Pandas DataFrame
28         The produced dataframe
29     """
30     classes_names = []
31     xml_list = []
32     for xml_file in glob.glob(path + "/*.xml"):
33         tree = ET.parse(xml_file)
34         root = tree.getroot()
35         for member in root.findall("object"):
```

## ❖ Useful commands in **add & commit**

```
git add scripts/app.js images/logo.png
```

```
git add *
```

```
git add .
```

```
git commit -a -m "MY MESSAGE HERE"
```

```
git push origin master<branch name>
```

```
git push origin HEAD:<branch>
```

```
Ex) $ git push origin HEAD:master ← 예전 버전: master
```

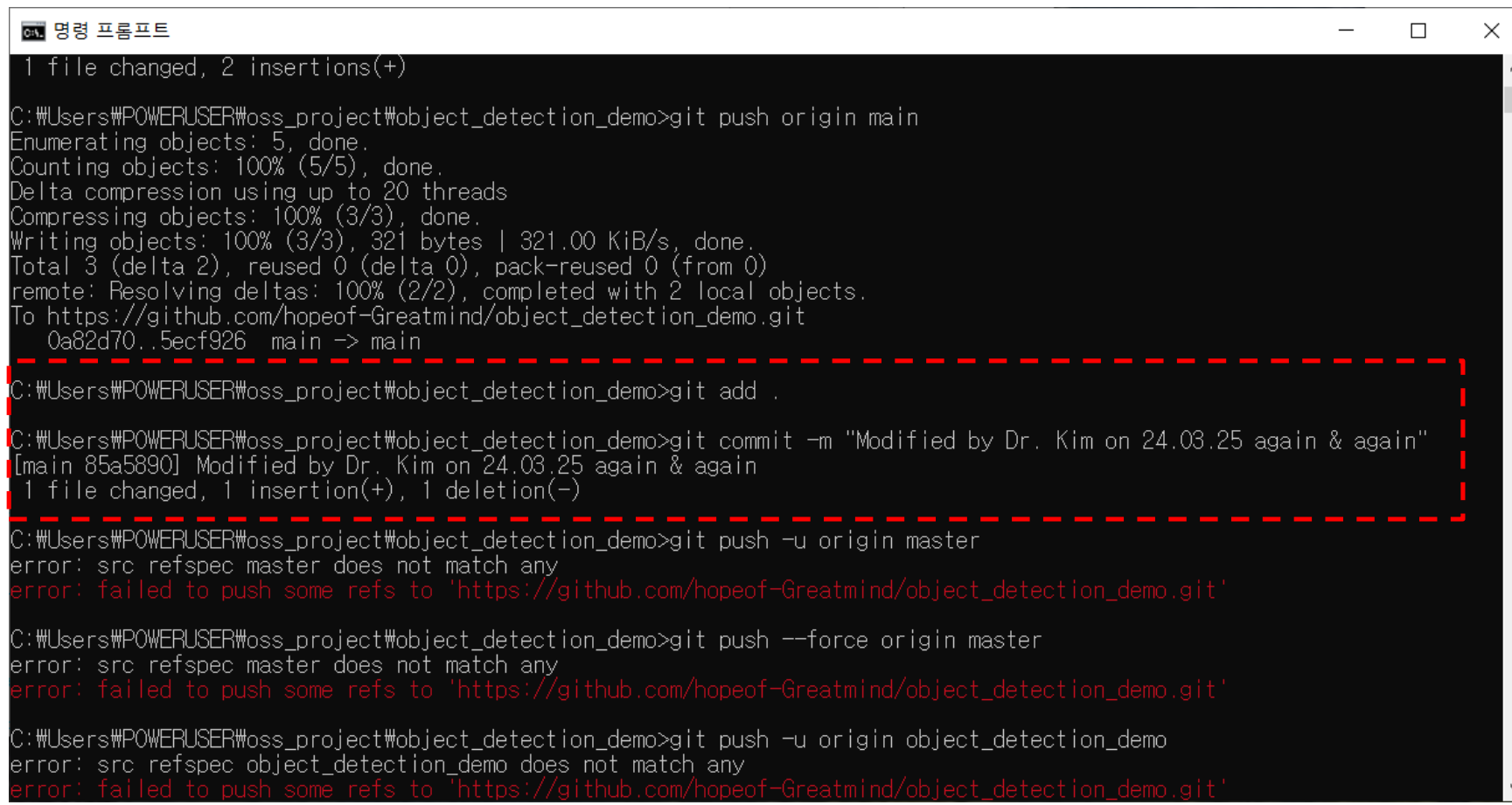


# How to use Git? (19)

- 9] change one source file and save...!!! Then using the following commands, we can push all changes to Github website.

```
$ git add .
```

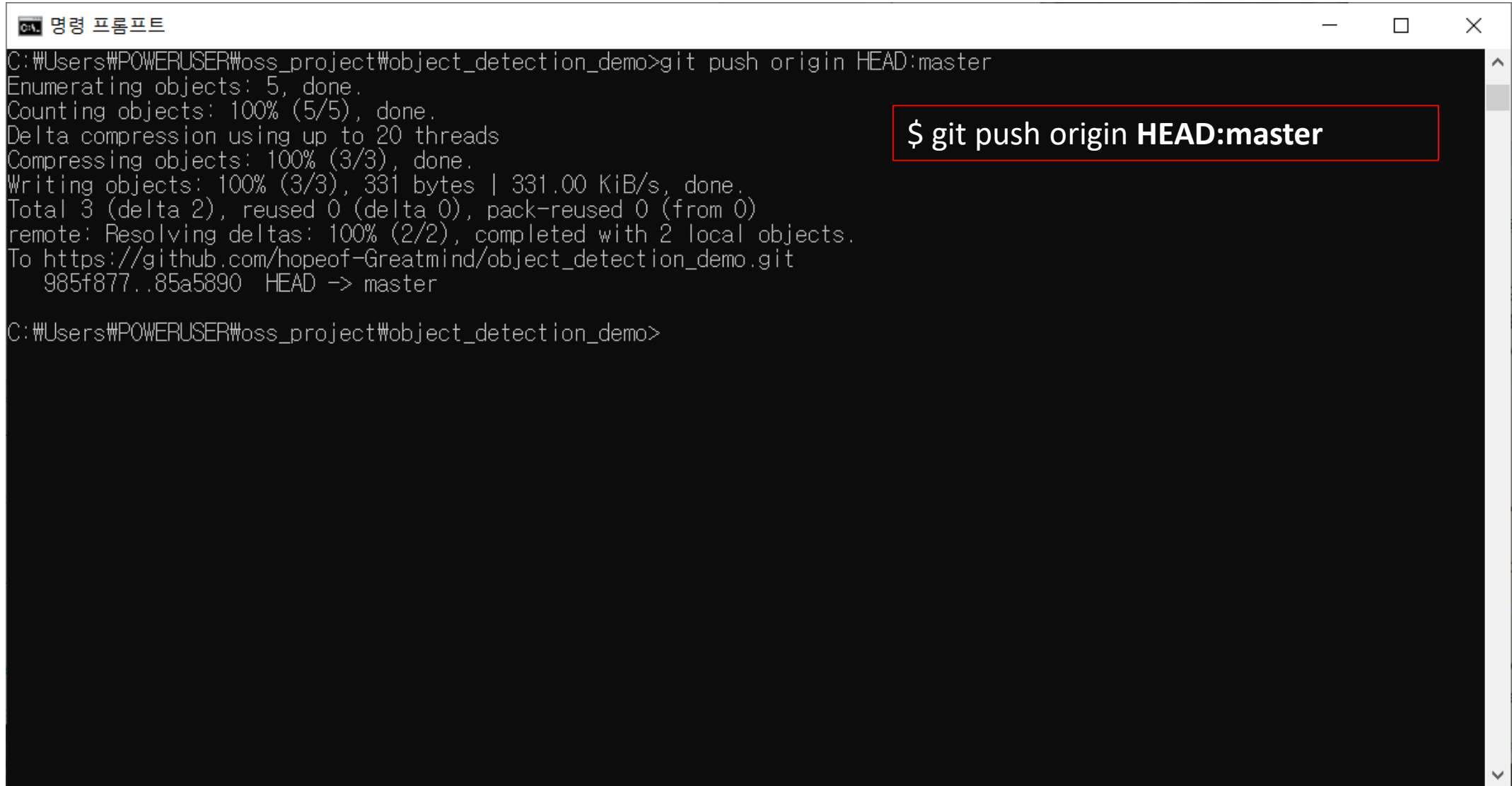
```
$ git commit -a -m "MY MESSAGE HERE"
```



```
명령 프롬프트
1 file changed, 2 insertions(+)
C:\Users\POWERUSER\oss_project\object_detection_demo>git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 20 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 321 bytes | 321.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/hopeof-Greatmind/object_detection_demo.git
   0a82d70..5ecf926  main -> main
C:\Users\POWERUSER\oss_project\object_detection_demo>git add .
C:\Users\POWERUSER\oss_project\object_detection_demo>git commit -m "Modified by Dr. Kim on 24.03.25 again & again"
[main 85a5890] Modified by Dr. Kim on 24.03.25 again & again
1 file changed, 1 insertion(+), 1 deletion(-)
C:\Users\POWERUSER\oss_project\object_detection_demo>git push -u origin master
error: src refspec master does not match any
error: failed to push some refs to 'https://github.com/hopeof-Greatmind/object_detection_demo.git'
C:\Users\POWERUSER\oss_project\object_detection_demo>git push --force origin master
error: src refspec master does not match any
error: failed to push some refs to 'https://github.com/hopeof-Greatmind/object_detection_demo.git'
C:\Users\POWERUSER\oss_project\object_detection_demo>git push -u origin object_detection_demo
error: src refspec object_detection_demo does not match any
error: failed to push some refs to 'https://github.com/hopeof-Greatmind/object_detection_demo.git'
```

# How to use Git? (20)

Then, you can push all changes to your Github site as:



```
명령 프롬프트
C:\Users\POWERUSER\oss_project\object_detection_demo>git push origin HEAD:master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 20 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 331 bytes | 331.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/hopeof-Greatmind/object_detection_demo.git
    985f877..85a5890 HEAD -> master

C:\Users\POWERUSER\oss_project\object_detection_demo>
```

\$ git push origin HEAD:master

# How to use Git? (21)

In your Github site and master branch, you can see the changed files and messages as:

The screenshot displays the GitHub interface for the repository 'object\_detection\_demo'. The 'master' branch is selected, showing a list of files and their recent changes. The 'Readme.md' and 'xml\_to\_csv.py' files are highlighted in yellow, indicating they have been modified. The 'Readme.md' file shows a commit message 'description of readme file' and a commit time of '2 hours ago'. The 'xml\_to\_csv.py' file shows a commit message 'Modified by Dr. Kim on 24.03.25 again & again' and a commit time of '22 minutes ago'. The repository also shows a README section with the text 'This is test file for OSS class in Sookmyung Women's University.' and a list of suggested workflows including Pylint and SLSA Generic generator.

object\_detection\_demo Public

Pin Unwatch 1 Fork 0 Star 0

master 2 Branches 0 Tags Go to file Add file <> Code About

hopeof-Greatmind Modified by Dr. Kim on 24.03.25 again & again 85a5890 · 22 minutes ago 35 Commits

data Delete IMG\_2383.JPG 5 years ago

deploy Add files via upload 5 years ago

test Add files via upload 5 years ago

Readme.md description of readme file 2 hours ago

export\_inference\_graph.py Add files via upload 5 years ago

generate\_tfrecord.py Add files via upload 5 years ago

local\_inference\_test.ipynb Add files via upload 5 years ago

local\_inference\_test.py Add files via upload 5 years ago

requirements.txt Add files via upload 5 years ago

resize\_images.py Add files via upload 5 years ago

tensorflow\_object\_detection\_training\_colab.ipynb Add files via upload 5 years ago

tensorflow\_object\_detection\_training\_colab\_ip... Colaboratory를 통해 생성됨 5 years ago

xml\_to\_csv.py Modified by Dr. Kim on 24.03.25 again & again 22 minutes ago

README

This is test file for OSS class in Sookmyung Women's University.

Releases

No releases published  
[Create a new release](#)

Packages

No packages published  
[Publish your first package](#)

Languages

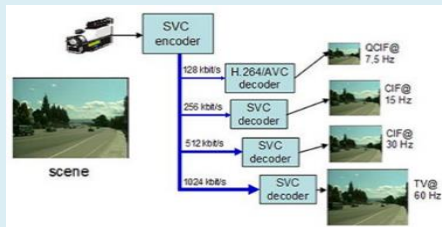
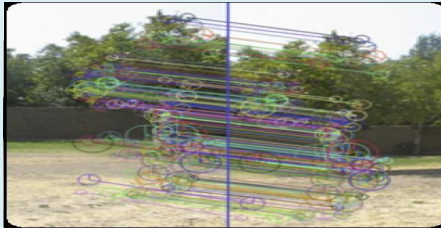
Jupyter Notebook 99.5% Python 0.5%

Suggested workflows

Based on your tech stack

Pylint Configure  
Lint a Python application with pylint.

SLSA Generic generator Configure



## Contents

---

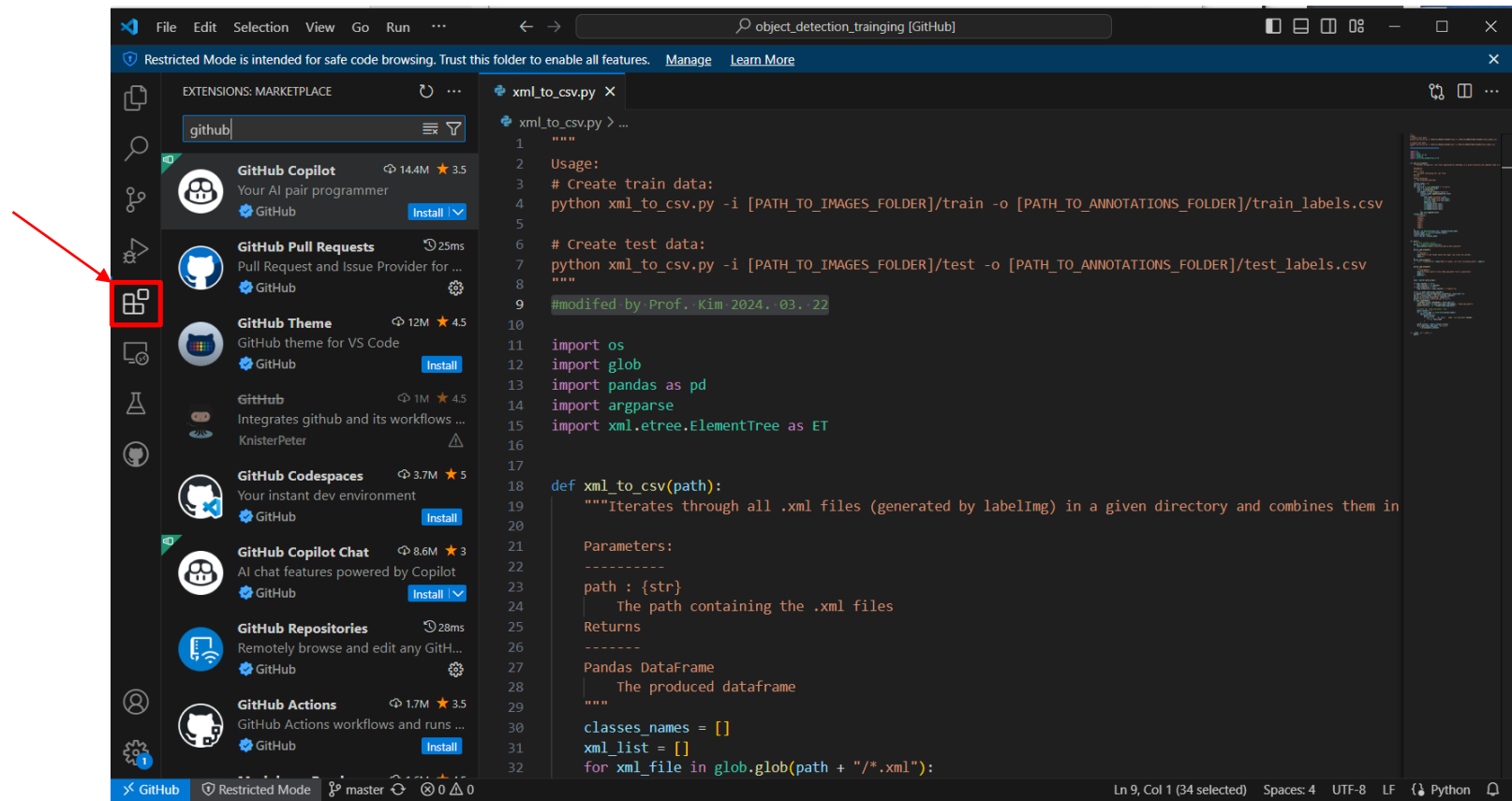
- How to use Git?
- Github + VS Code

## ❖ Requirements

- Install your VS code at [Download Visual Studio Code - Mac, Linux, Windows.](#)

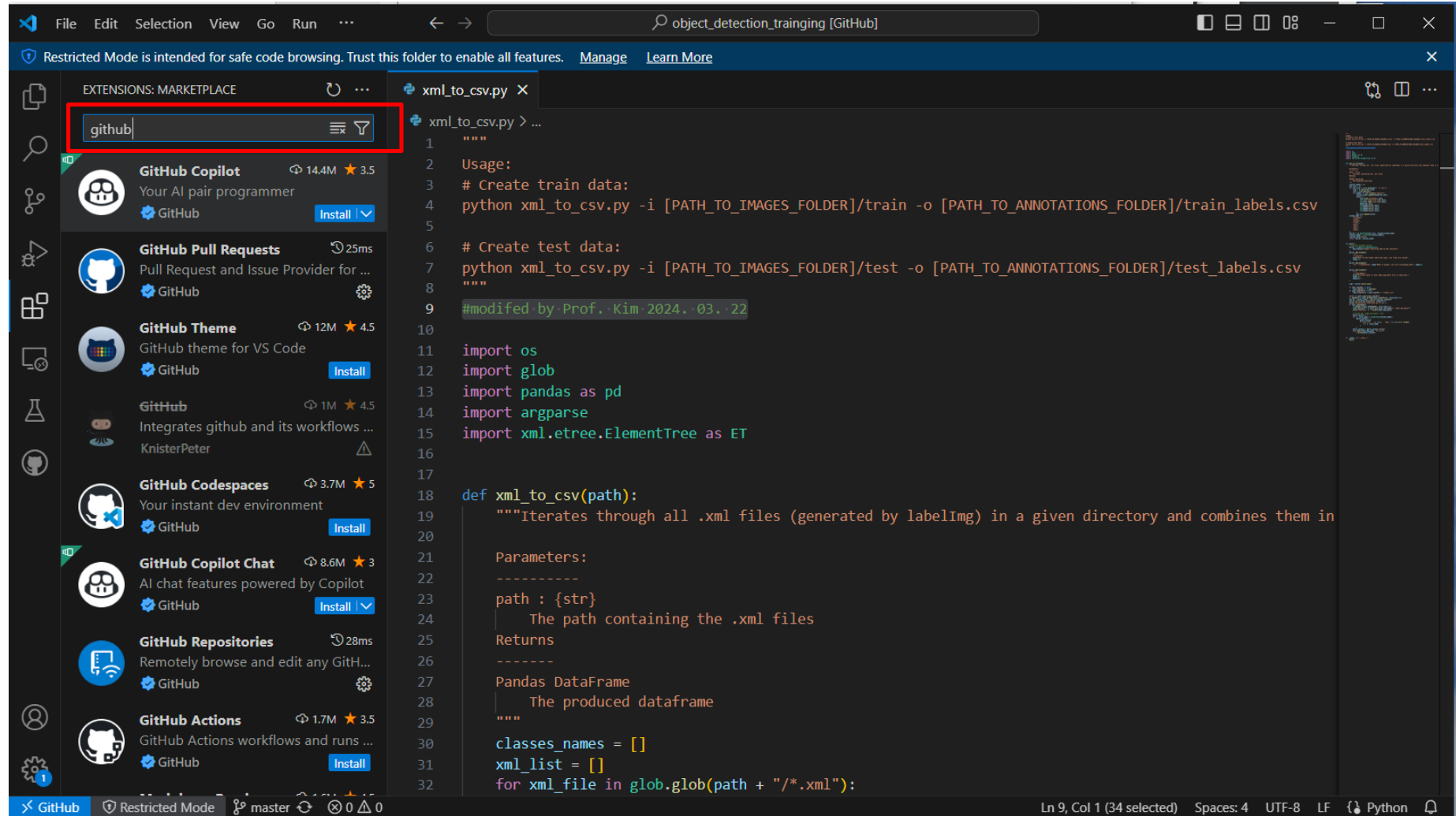
## ❖ Steps to Manage Your Github source in Your VS code

- 2] Open your VS code.
- 3] Click "Extension" tab on your VS code program!



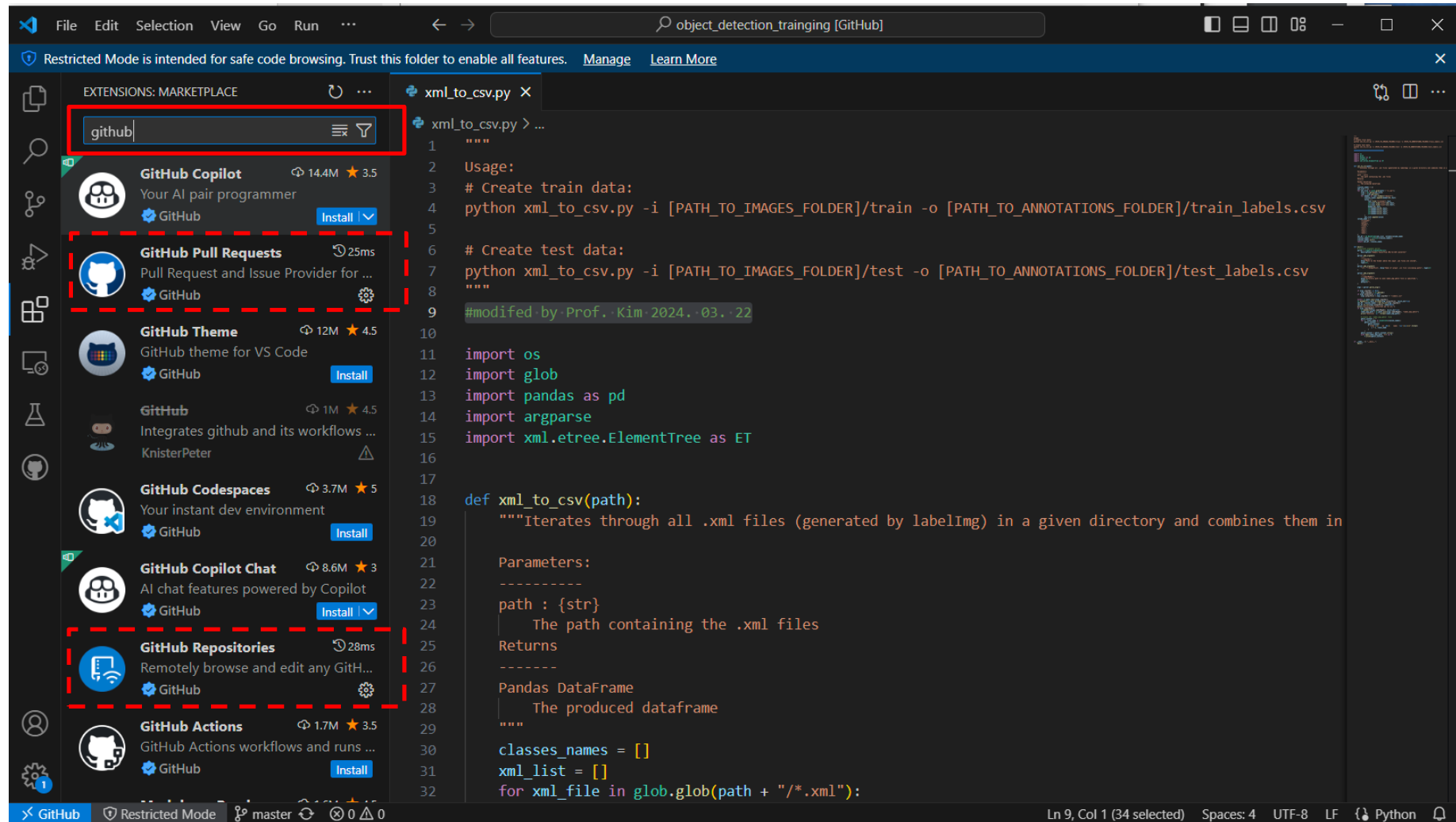
# Github + VS Code (2)

- 4] Search Github extension in "search field". You can see Github extension programs...!!!

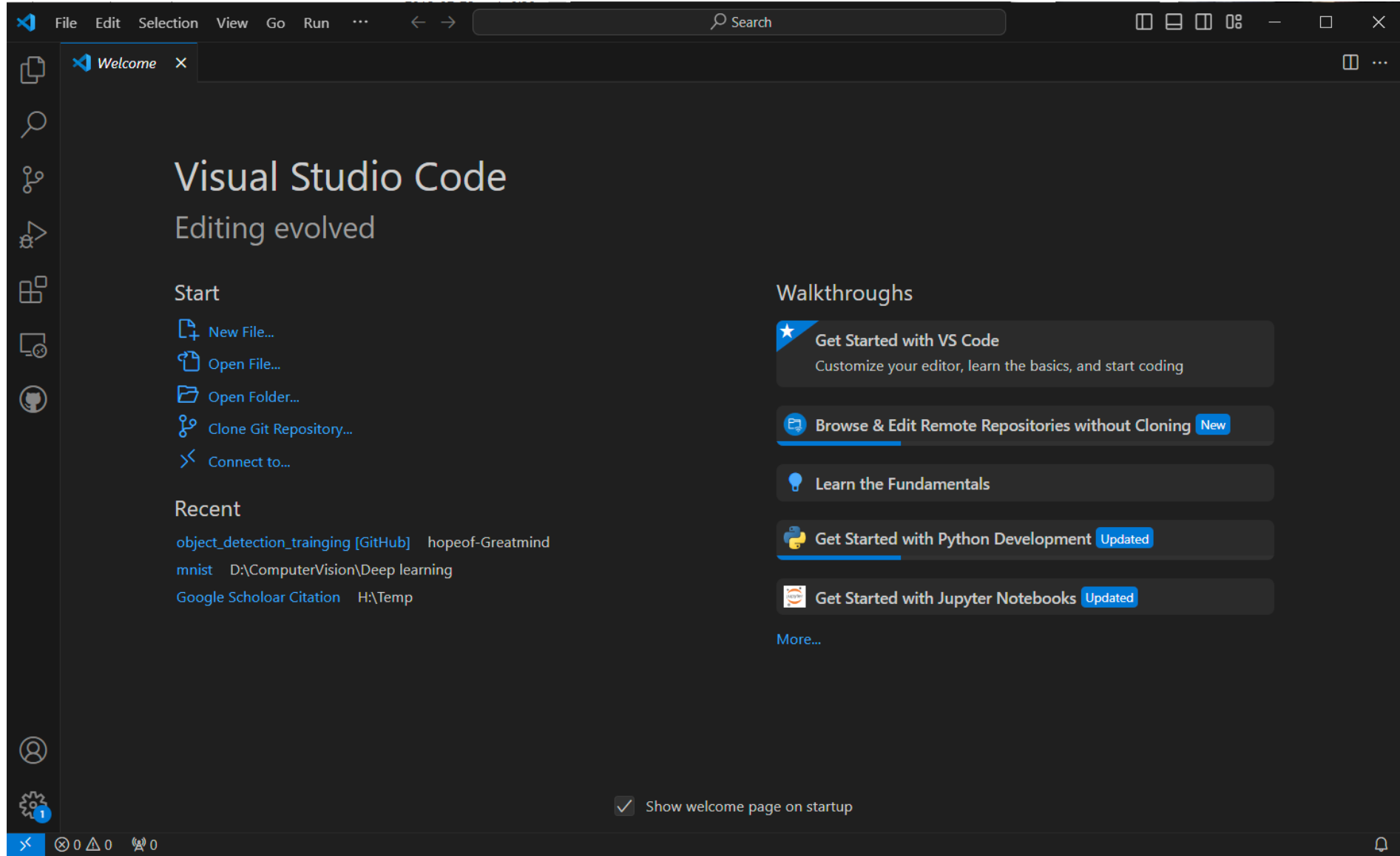


# Github + VS Code (3)

- 5] Install the required packages by selecting them.



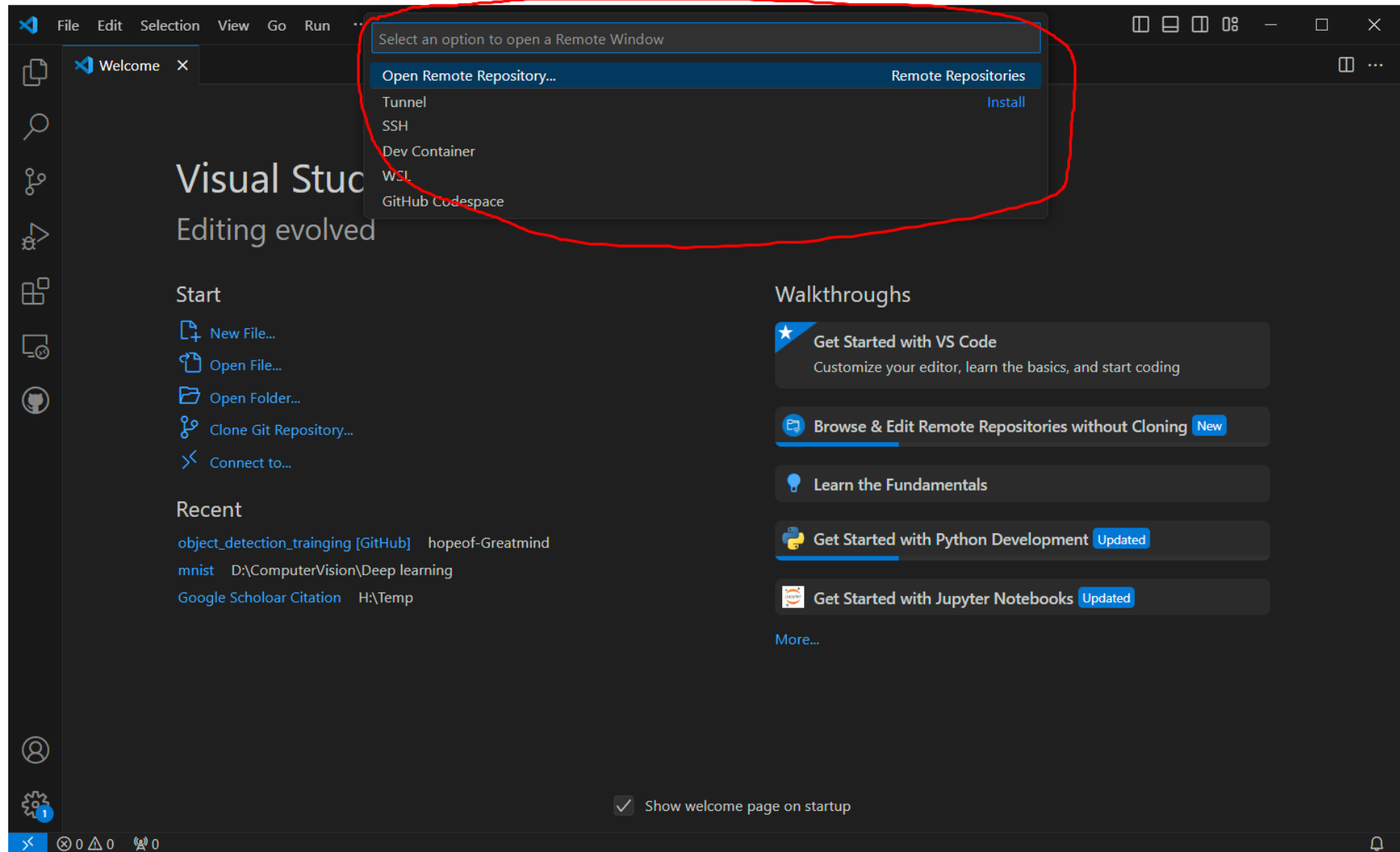
- 6] In staring page of VS code, select start item as "Connect to..."





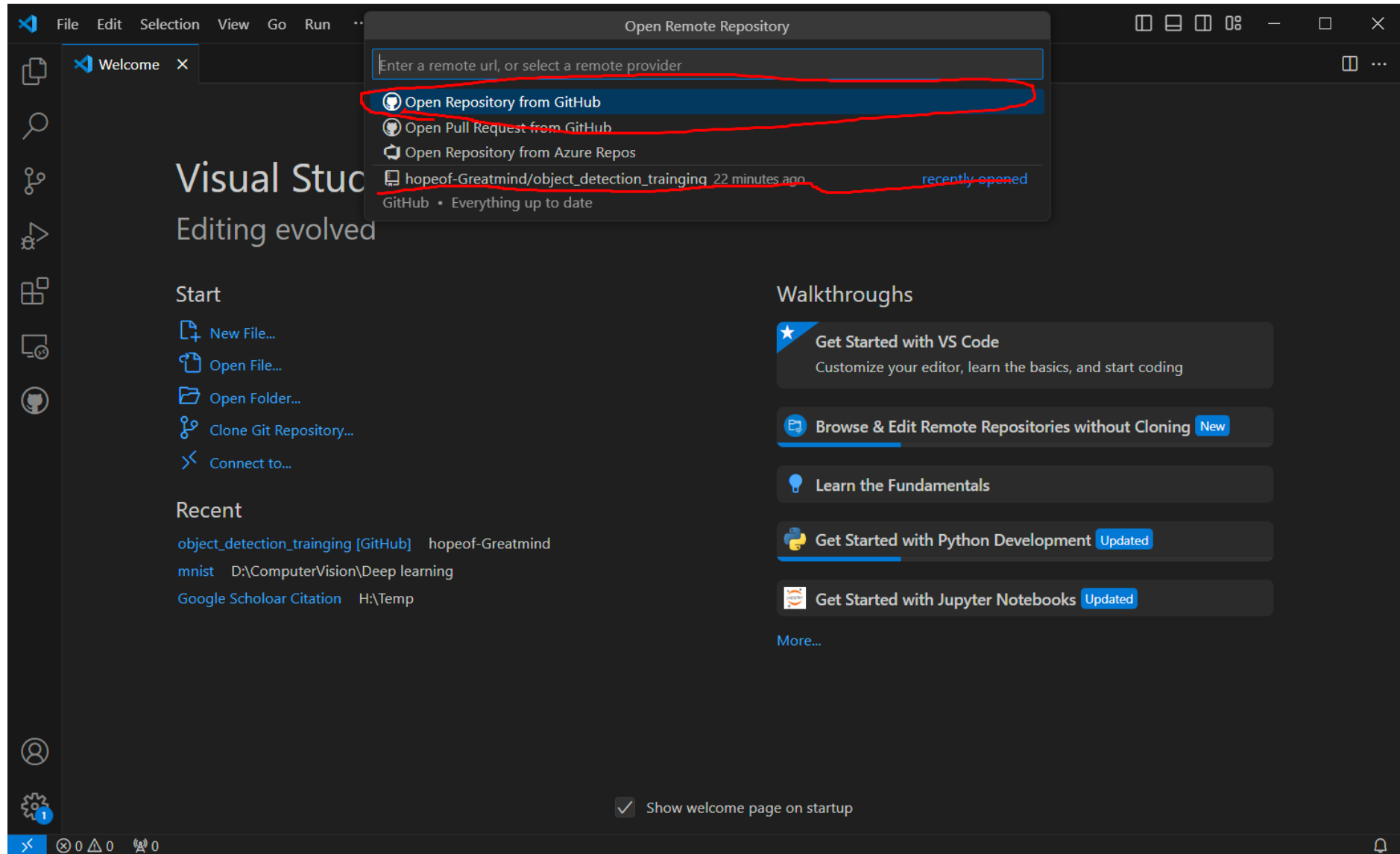
# Github + VS Code (5)

- Select "Open Remote Repository"



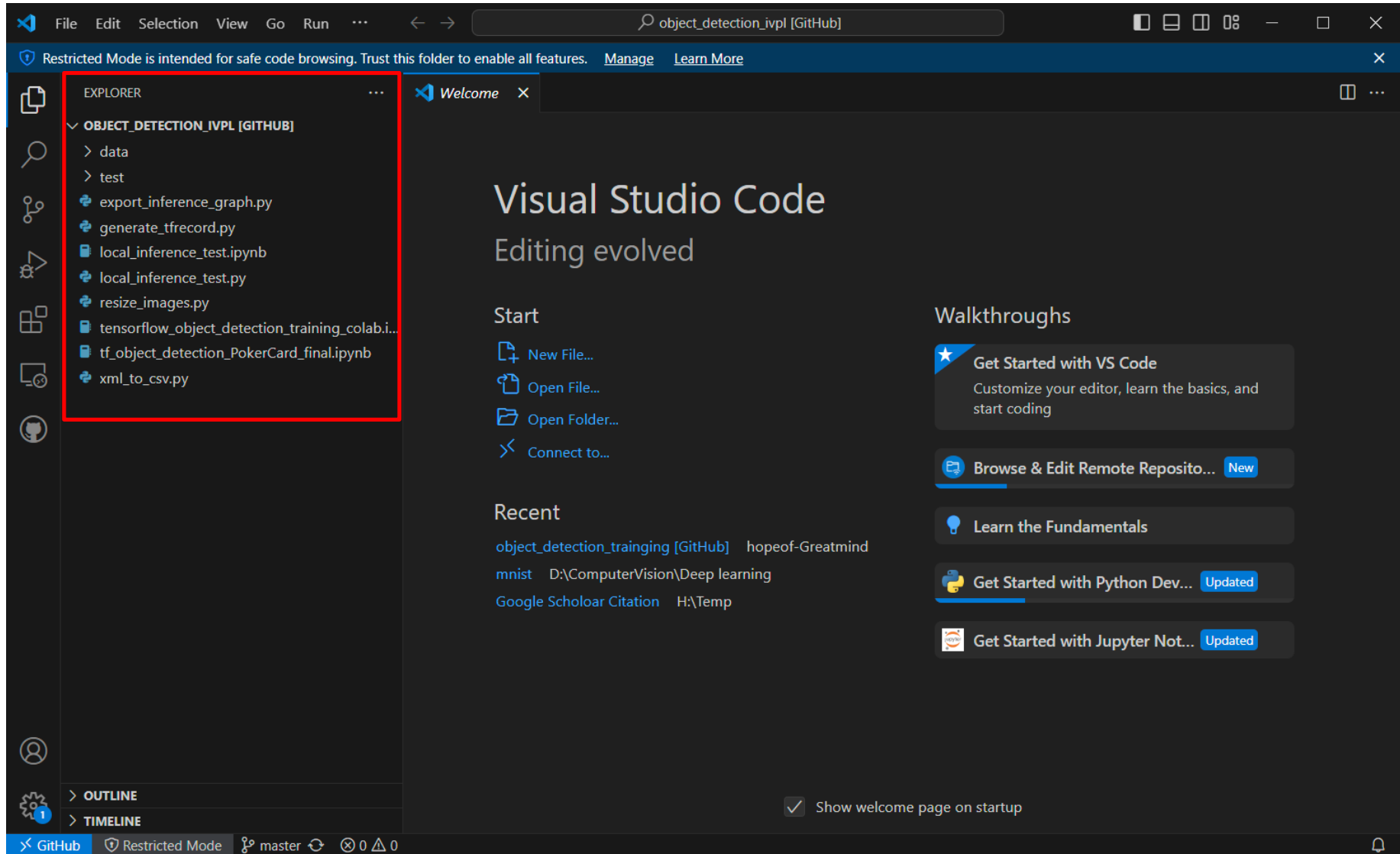
# Github + VS Code (6)

- Select "Open Repository from Github."



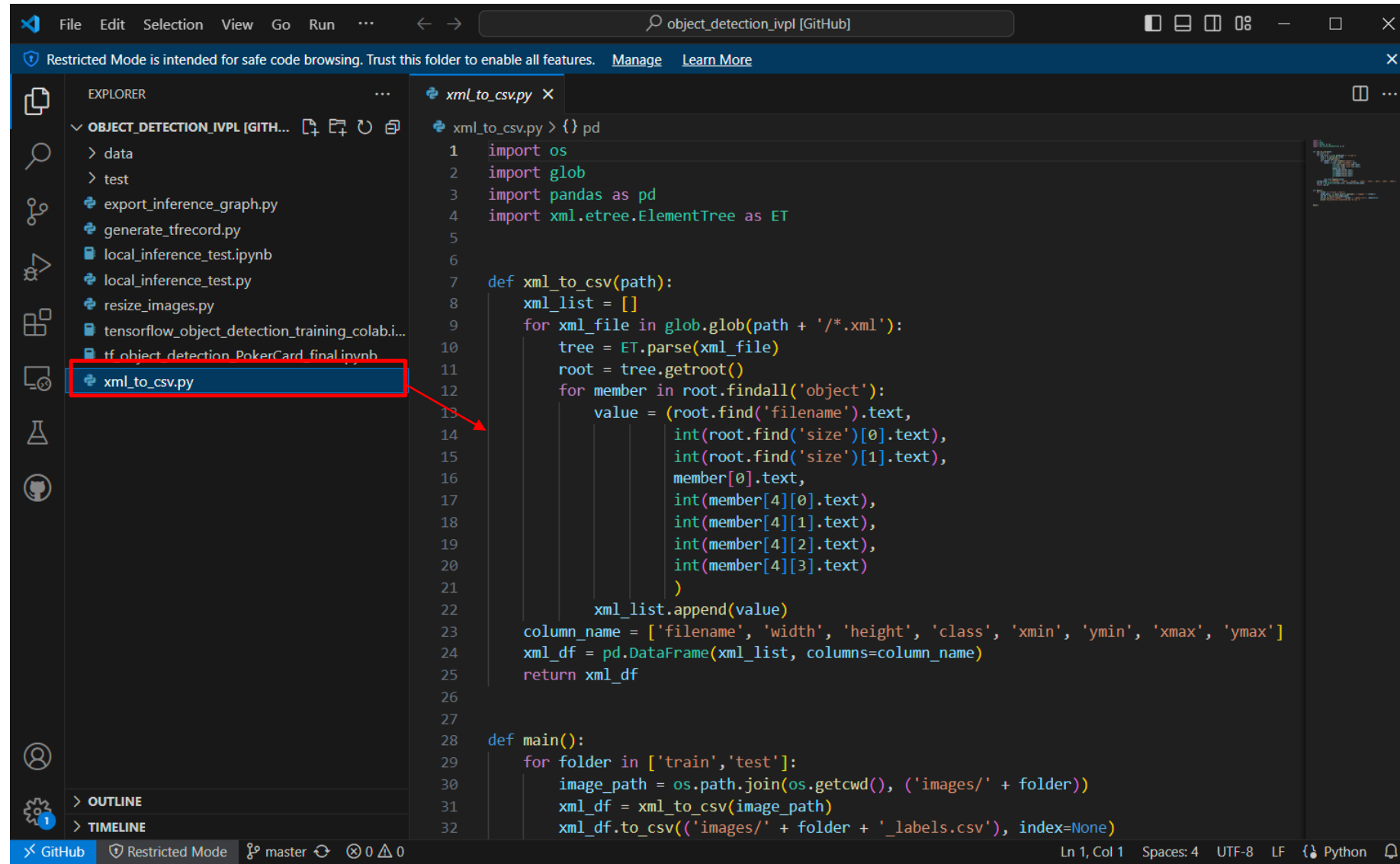
# Github + VS Code (7)

- Then you can see the log-in page at Github. Just log-in (sign-in) on Github.
- Then, you see the work-folder selection for your job.



# Github + VS Code (8)

- 7] Select one source file to modify the content (or program). In our example, I will select "xml\_to\_csv.py" file. Just click "xml\_to\_csv.py" file..!!!



The screenshot shows the Visual Studio Code interface. In the Explorer sidebar on the left, the file 'xml\_to\_csv.py' is highlighted with a red rectangle. An arrow points from this rectangle to the same file, which is now open in the main editor window. The editor displays the Python code for 'xml\_to\_csv.py', which includes imports for 'os', 'glob', 'pandas', and 'xml.etree.ElementTree', a function 'xml\_to\_csv' that processes XML files into a pandas DataFrame, and a 'main' function that iterates over 'train' and 'test' folders to process images.

```
1 import os
2 import glob
3 import pandas as pd
4 import xml.etree.ElementTree as ET
5
6
7 def xml_to_csv(path):
8     xml_list = []
9     for xml_file in glob.glob(path + '/*.xml'):
10         tree = ET.parse(xml_file)
11         root = tree.getroot()
12         for member in root.findall('object'):
13             value = (root.find('filename').text,
14                     int(root.find('size')[0].text),
15                     int(root.find('size')[1].text),
16                     member[0].text,
17                     int(member[4][0].text),
18                     int(member[4][1].text),
19                     int(member[4][2].text),
20                     int(member[4][3].text)
21                     )
22             xml_list.append(value)
23     column_name = ['filename', 'width', 'height', 'class', 'xmin', 'ymin', 'xmax', 'ymax']
24     xml_df = pd.DataFrame(xml_list, columns=column_name)
25     return xml_df
26
27
28 def main():
29     for folder in ['train', 'test']:
30         image_path = os.path.join(os.getcwd(), ('images/' + folder))
31         xml_df = xml_to_csv(image_path)
32         xml_df.to_csv('images/' + folder + '_labels.csv', index=None)
```

# Github + VS Code (9)

- 8] Edit some part or add additional code (comment) line. And save it.

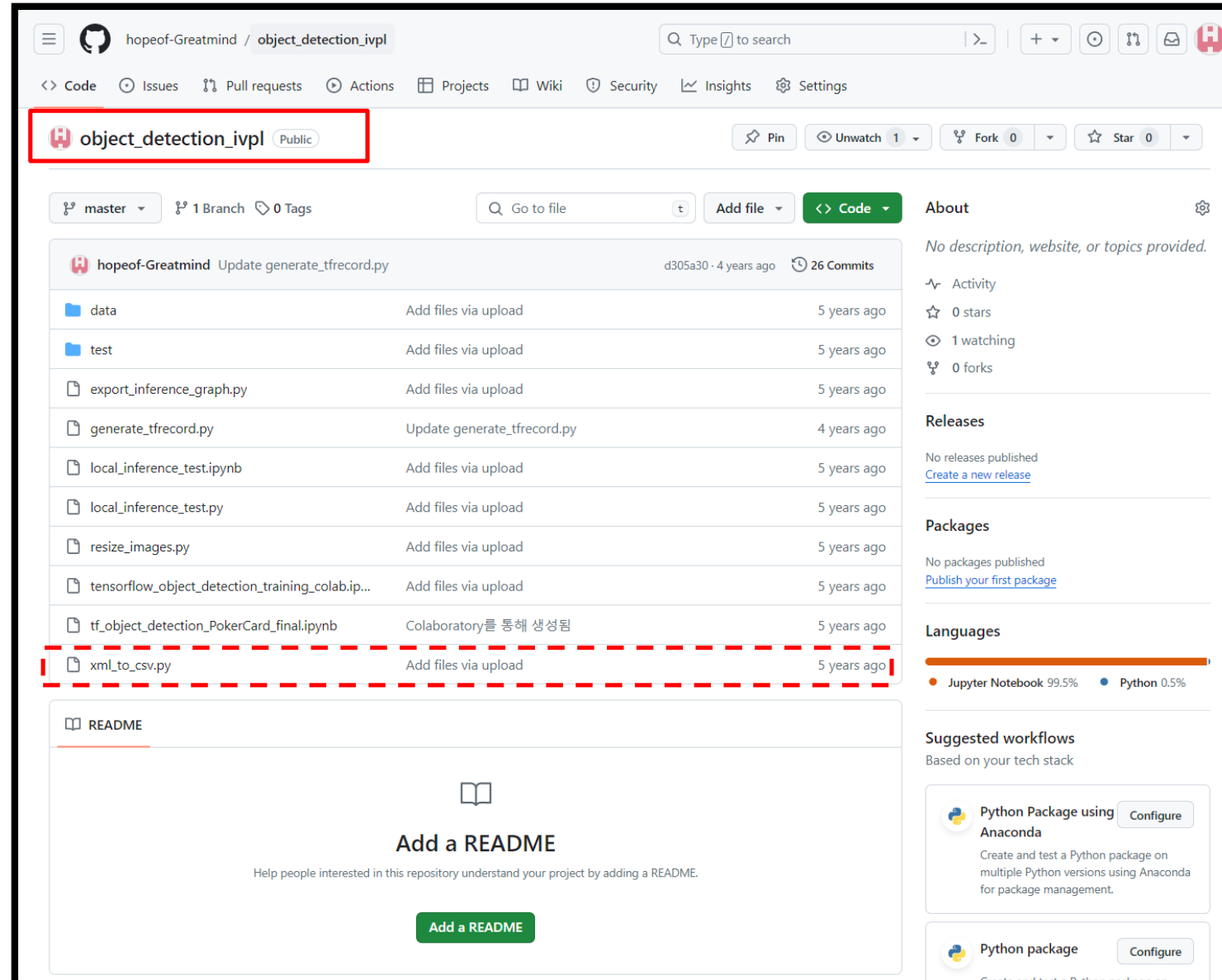
```
File Edit Selection View Go Run ... object_detection_ivpl [GitHub]
Restricted Mode is intended for safe code browsing. Trust this folder to enable all features. Manage Learn More

EXPLORER
OBJECT_DETECTION_IVPL [GITHUB]
  data
  test
  export_inference_graph.py
  generate_tfrecord.py
  local_inference_test.ipynb
  local_inference_test.py
  resize_images.py
  tensorflow_object_detection_training_colab.i...
  tf_object_detection_PokerCard_final.ipynb
  xml_to_csv.py

xml_to_csv.py
1 ''' Modified by Prof. Kim on 2024. 03. 22 for our great student
2 in Sookmyung Women's Univeristy'''
3
4 import os
5 import glob
6 import pandas as pd
7 import xml.etree.ElementTree as ET
8
9
10 def xml_to_csv(path):
11     xml_list = []
12     for xml_file in glob.glob(path + '/*.xml'):
13         tree = ET.parse(xml_file)
14         root = tree.getroot()
15         for member in root.findall('object'):
16             value = (root.find('filename').text,
17                     int(root.find('size')[0].text),
18                     int(root.find('size')[1].text),
19                     member[0].text,
20                     int(member[4][0].text),
21                     int(member[4][1].text),
22                     int(member[4][2].text),
23                     int(member[4][3].text)
24                     )
25             xml_list.append(value)
26     column_name = ['filename', 'width', 'height', 'class', 'xmin', 'ymin', 'xmax', 'ymax']
27     xml_df = pd.DataFrame(xml_list, columns=column_name)
28     return xml_df
29
30
31 def main():
32     for folder in ['train', 'test']:
```

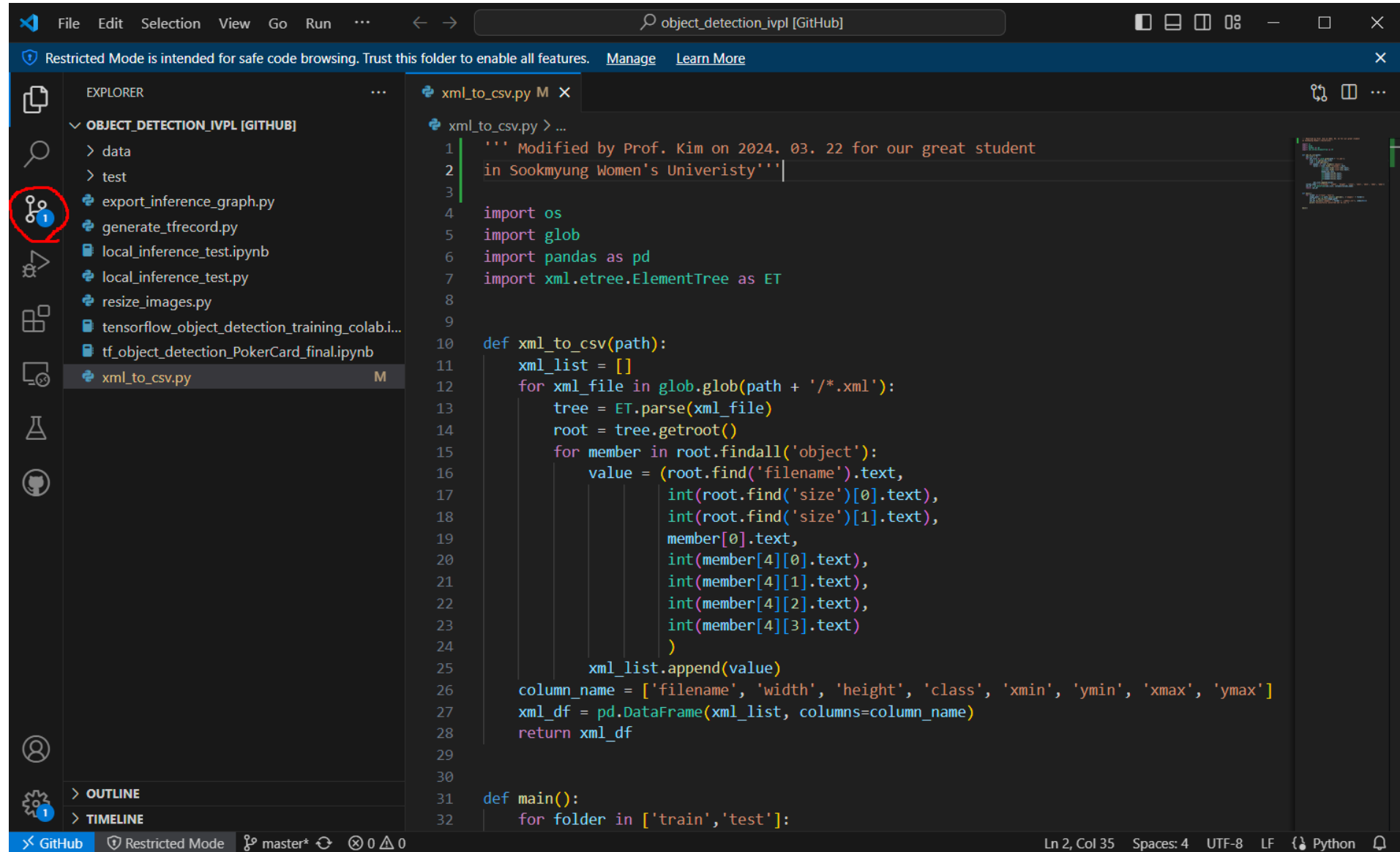
# Github + VS Code (10)

- 9] Go to your Github (work-folder).



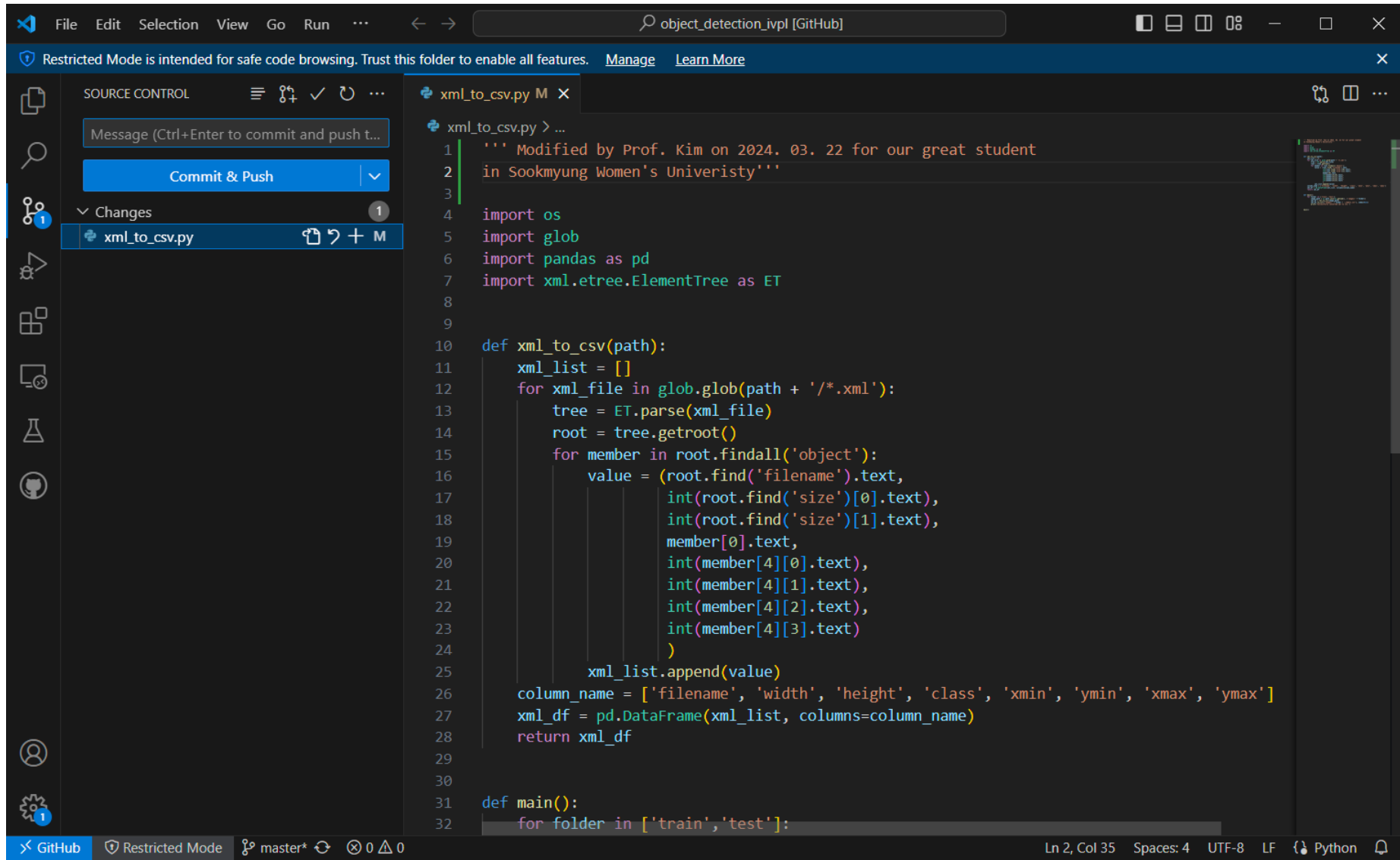
# Github + VS Code (11)

- 10] Again go back to VS code and select "Source control" tab in left.



# Github + VS Code (12)

- 11] Again go back to VS code and select "Source control" tab in left. You can see "message field" and "Commit & Push" button!

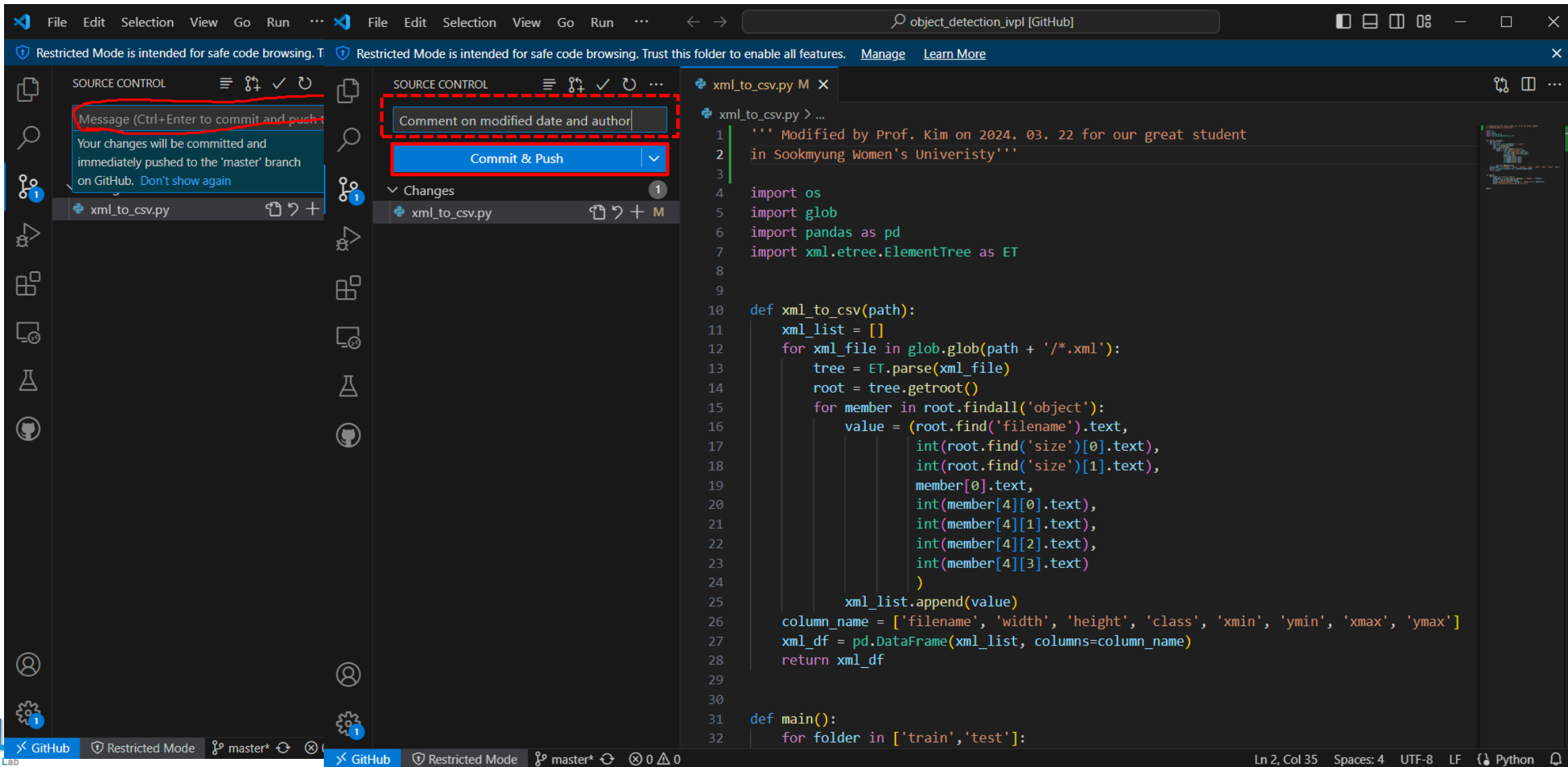
The screenshot shows the Visual Studio Code interface with the Source Control view on the left. The 'Commit & Push' button is highlighted in blue. The main editor displays the code for 'xml\_to\_csv.py'.

```
1 ''' Modified by Prof. Kim on 2024. 03. 22 for our great student
2 in Sookmyung Women's Univeristy'''
3
4 import os
5 import glob
6 import pandas as pd
7 import xml.etree.ElementTree as ET
8
9
10 def xml_to_csv(path):
11     xml_list = []
12     for xml_file in glob.glob(path + '/*.xml'):
13         tree = ET.parse(xml_file)
14         root = tree.getroot()
15         for member in root.findall('object'):
16             value = (root.find('filename').text,
17                     int(root.find('size')[0].text),
18                     int(root.find('size')[1].text),
19                     member[0].text,
20                     int(member[4][0].text),
21                     int(member[4][1].text),
22                     int(member[4][2].text),
23                     int(member[4][3].text)
24                     )
25             xml_list.append(value)
26     column_name = ['filename', 'width', 'height', 'class', 'xmin', 'ymin', 'xmax', 'ymax']
27     xml_df = pd.DataFrame(xml_list, columns=column_name)
28     return xml_df
29
30
31 def main():
32     for folder in ['train', 'test']:
```



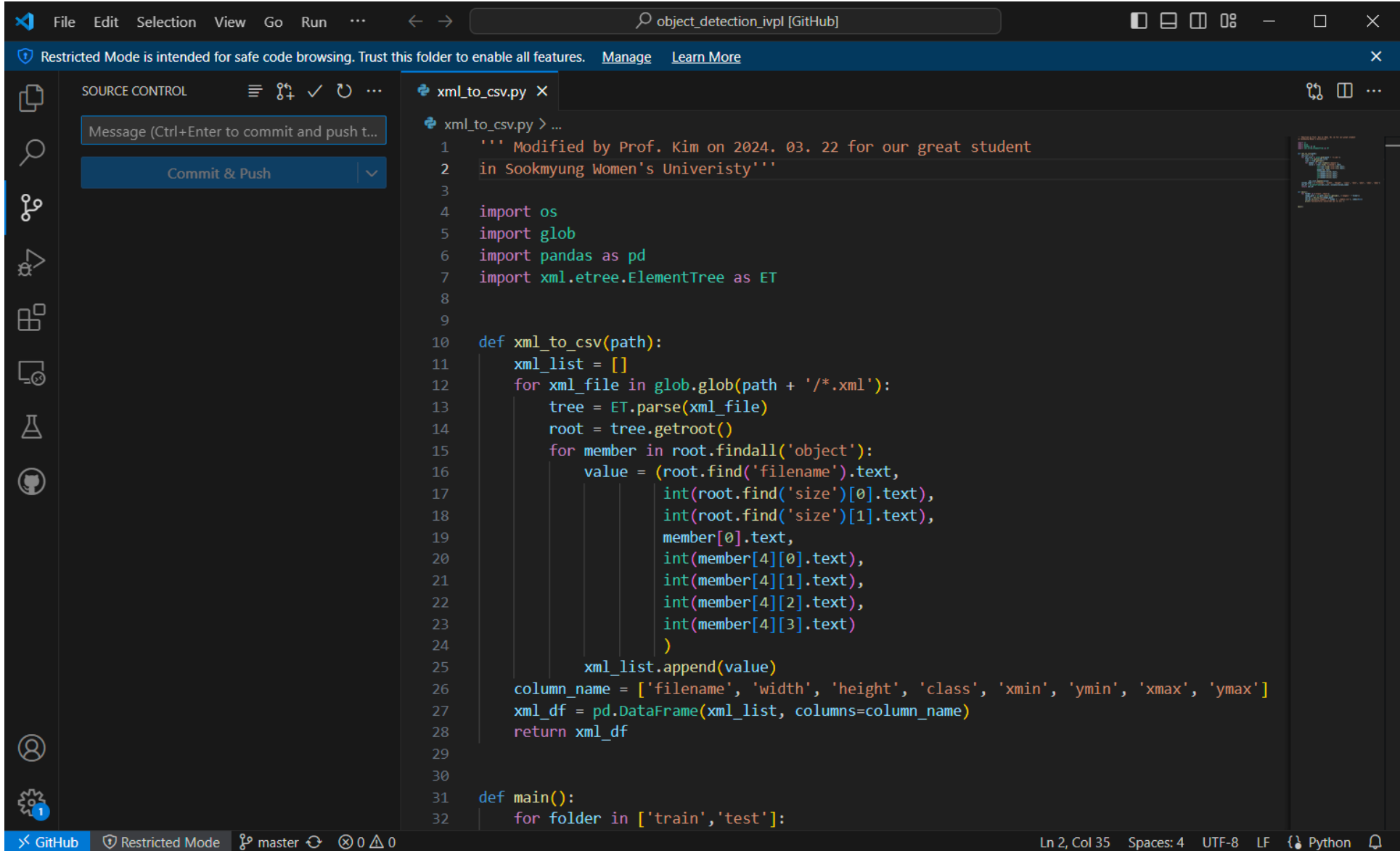
# Github + VS Code (13)

- 12] First, give the message for commit and click "Commit & Push" button!



# Github + VS Code (14)

- You can see the change file was disappeared.



```
object_detection_ivpl [GitHub]
Restricted Mode is intended for safe code browsing. Trust this folder to enable all features. Manage Learn More

SOURCE CONTROL
Message (Ctrl+Enter to commit and push t...
Commit & Push

xml_to_csv.py X
xml_to_csv.py > ...
1 ''' Modified by Prof. Kim on 2024. 03. 22 for our great student
2 in Sookmyung Women's Univeristy'''
3
4 import os
5 import glob
6 import pandas as pd
7 import xml.etree.ElementTree as ET
8
9
10 def xml_to_csv(path):
11     xml_list = []
12     for xml_file in glob.glob(path + '/*.xml'):
13         tree = ET.parse(xml_file)
14         root = tree.getroot()
15         for member in root.findall('object'):
16             value = (root.find('filename').text,
17                     int(root.find('size')[0].text),
18                     int(root.find('size')[1].text),
19                     member[0].text,
20                     int(member[4][0].text),
21                     int(member[4][1].text),
22                     int(member[4][2].text),
23                     int(member[4][3].text)
24                     )
25             xml_list.append(value)
26     column_name = ['filename', 'width', 'height', 'class', 'xmin', 'ymin', 'xmax', 'ymax']
27     xml_df = pd.DataFrame(xml_list, columns=column_name)
28     return xml_df
29
30
31 def main():
32     for folder in ['train', 'test']:
```

Ln 2, Col 35 Spaces: 4 UTF-8 LF Python

# Github + VS Code (15)

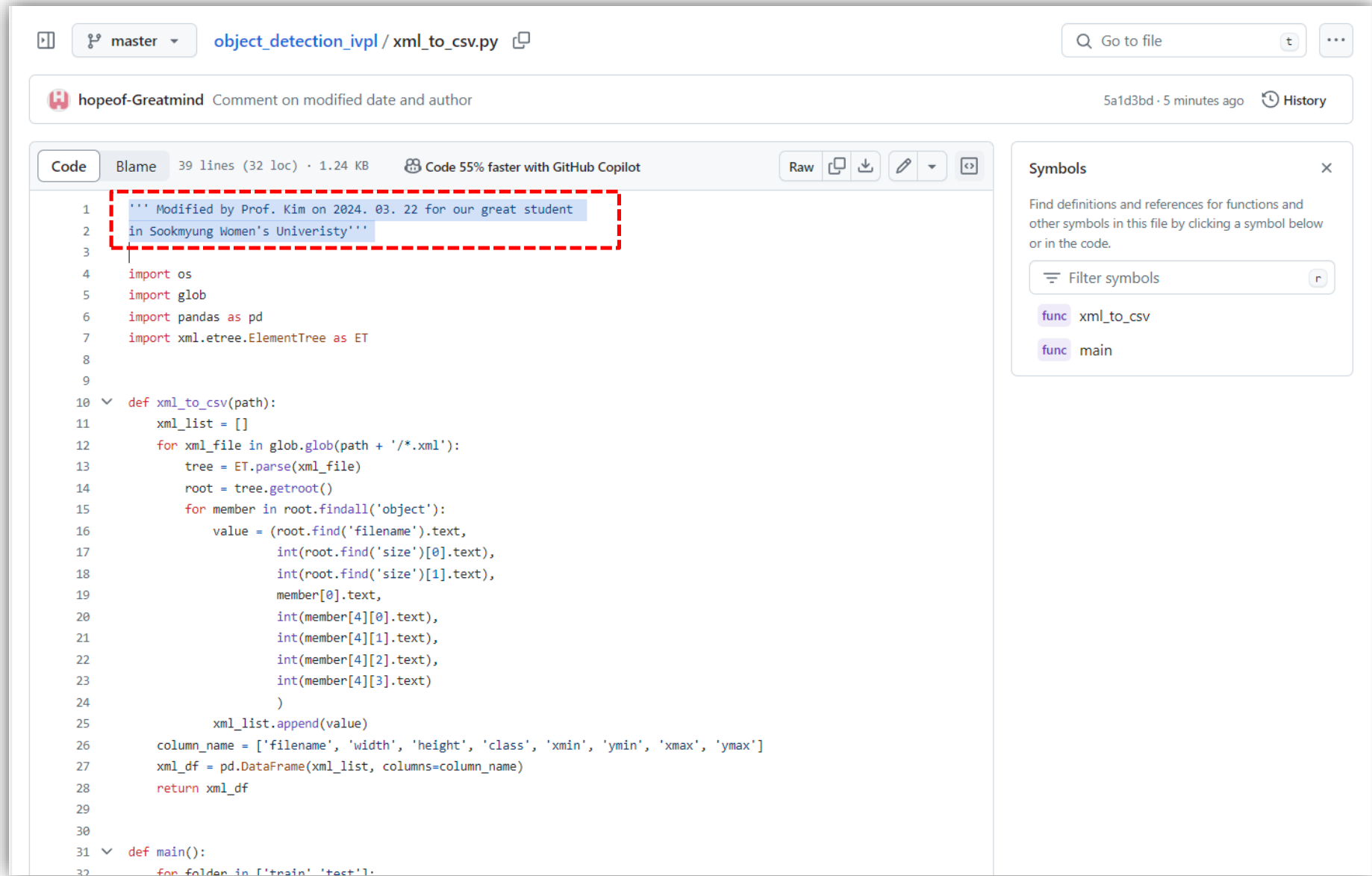
- 13] Then let's go to Github site and check the change of yours...!

The screenshot shows the GitHub interface for the repository 'object\_detection\_ivpl' by user 'hopeof-Greatmind'. The repository is public and has 1 branch (master) and 0 tags. The file list shows several files, with 'xml\_to\_csv.py' highlighted by a red dashed box. The file 'xml\_to\_csv.py' was modified 2 minutes ago. The repository also has a README section with a button to 'Add a README'.

File Name	Action	Time
data	Add files via upload	5 years ago
test	Add files via upload	5 years ago
export_inference_graph.py	Add files via upload	5 years ago
generate_tfrecord.py	Update generate_tfrecord.py	4 years ago
local_inference_test.ipynb	Add files via upload	5 years ago
local_inference_test.py	Add files via upload	5 years ago
resize_images.py	Add files via upload	5 years ago
tensorflow_object_detection_training_colab.ip...	Add files via upload	5 years ago
tf_object_detection_PokerCard_final.ipynb	Colaboratory를 통해 생성됨	5 years ago
xml_to_csv.py	Comment on modified date and author	2 minutes ago

# Github + VS Code (16)

- Just click "xml\_to\_csv.py" to see the change which was reflected.

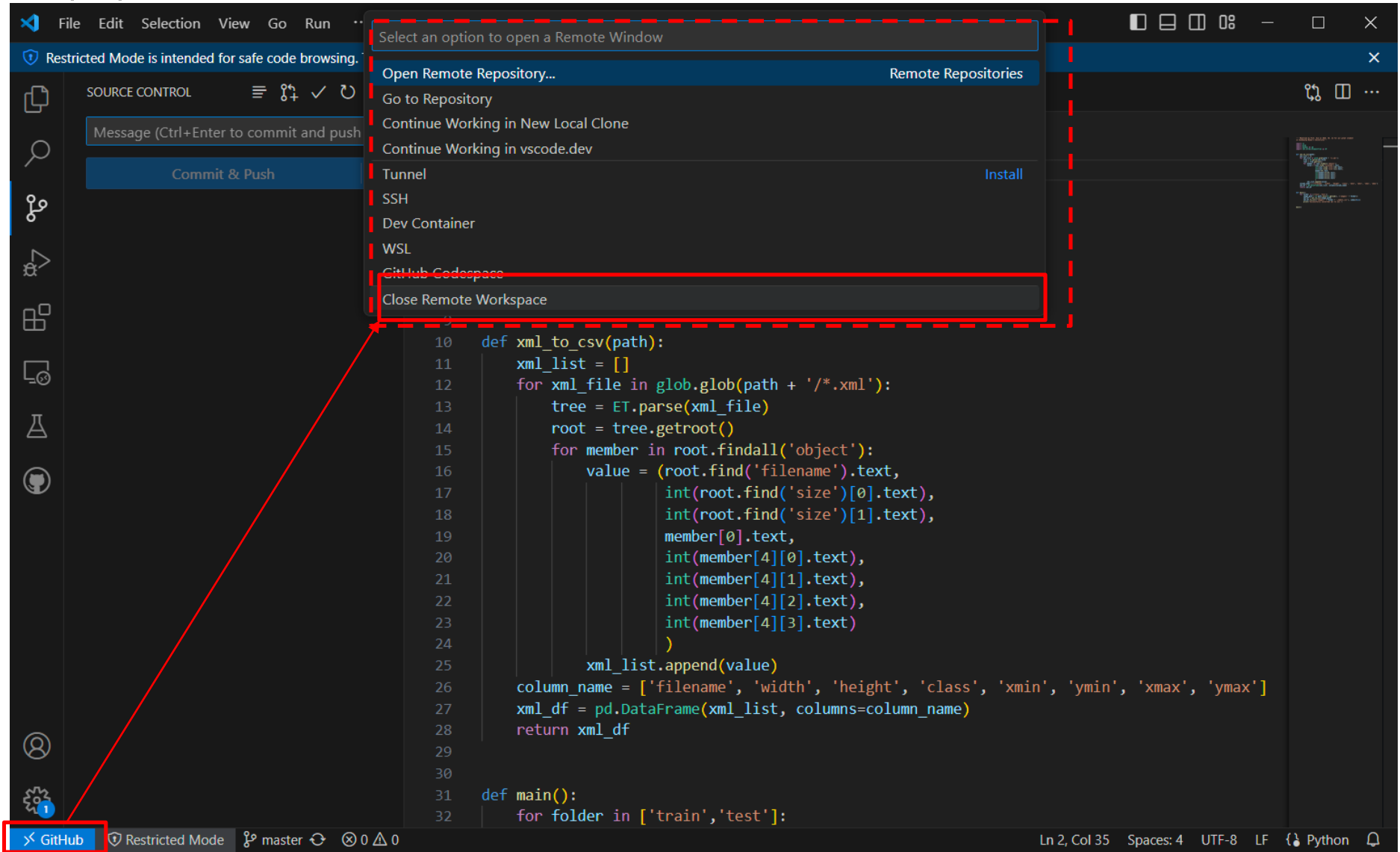


The screenshot displays the GitHub web interface for a file named `xml_to_csv.py` in the repository `object_detection_ivpl`. The file is currently on the `master` branch. The interface shows a diff view with a red dashed box highlighting the changes in the docstring at lines 1 and 2. The docstring is updated to reflect a modification by Prof. Kim on 2024.03.22 for a great student at Sookmyung Women's University. The code includes imports for `os`, `glob`, `pandas`, and `xml.etree.ElementTree`. It defines a function `xml_to_csv` that processes XML files into a pandas DataFrame and a `main` function. The right sidebar shows the Symbols panel, which lists the functions `xml_to_csv` and `main`.

```
1 ''' Modified by Prof. Kim on 2024. 03. 22 for our great student
2 in Sookmyung Women's Univeristy'''
3
4 import os
5 import glob
6 import pandas as pd
7 import xml.etree.ElementTree as ET
8
9
10 def xml_to_csv(path):
11     xml_list = []
12     for xml_file in glob.glob(path + '/*.xml'):
13         tree = ET.parse(xml_file)
14         root = tree.getroot()
15         for member in root.findall('object'):
16             value = (root.find('filename').text,
17                     int(root.find('size')[0].text),
18                     int(root.find('size')[1].text),
19                     member[0].text,
20                     int(member[4][0].text),
21                     int(member[4][1].text),
22                     int(member[4][2].text),
23                     int(member[4][3].text)
24             )
25             xml_list.append(value)
26     column_name = ['filename', 'width', 'height', 'class', 'xmin', 'ymin', 'xmax', 'ymax']
27     xml_df = pd.DataFrame(xml_list, columns=column_name)
28     return xml_df
29
30
31 def main():
32     for folder in ['train', 'test']:
```

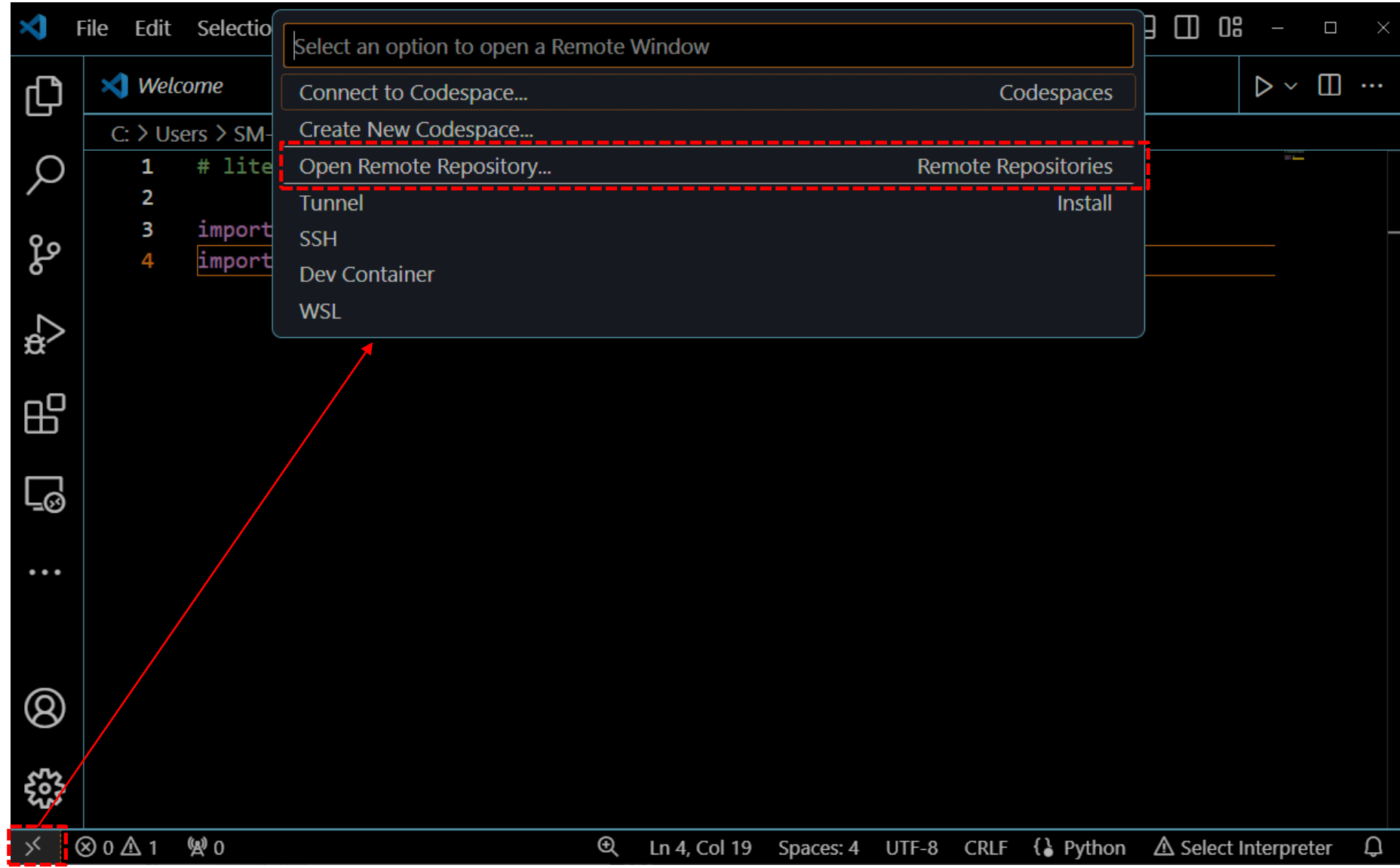
# Github + VS Code (17)

- 14] **In VS Code**, finish the Remote workspace: Just click  on left-bottom. Then you can see the pop-down items to close it.



# Github + VS Code (18)

- ❖ When VS code has some files at beginning, you can click the “github” icon at bottom-left.



**Thank you for your attention.!!!**  
**QnA**

<http://ivpl.sookmyung.ac.kr>