Create a new directory called `ml-project`

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets> mkdir "ml-project"

Directory: S:\Practice Python yourself\AI projects\Resources_Notes_datasheets


```
Mode            LastWriteTime        Length Name
----            -------------        ------ ----
d-----      18-02-2026    17:26             ml-project
```

Navigate into that directory

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets> cd .\ml-project\

Initialize a Git repository

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git init

Initialized empty Git repository in S:/Practice Python yourself/AI projects/Resources_Notes_datasheets/ml-project/.git/

Create four files

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> echo "I will learn Git today" > train.py

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> echo "I will learn Git today" > predict.py

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> echo "I will learn Git today" > utils.py

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> echo "I will learn Git today" > README.md

```
PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
LS

Directory: S:\Practice Python yourself\AI
projects\Resources_Notes_datasheets\ml-project

Mode              LastWriteTime         Length Name

----              -------------         ------ ----

-a----      18-02-2026    19:06          50 predict.py

-a----      18-02-2026    19:06          50 README.md

-a----      18-02-2026    19:03          50 train.py

-a----      18-02-2026    19:06          50 utils.py
```

Check the status of your Git repository

```
PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
git status

On branch master

No commits yet

Untracked files:

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

        README.md

        predict.py

        train.py

        utils.py

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

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
git add train.py

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
git add utils.py

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
git status

On branch master

No commits yet

Changes to be committed:

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

    new file: train.py

    new file: utils.py

Untracked files:

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

    README.md

    predict.py

Commit these changes with the message "Add training script and utilities"

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git commit -m "Adding training scripts and utilities"

[master (root-commit) 67bdd0b] Adding training scripts and utilities

 2 files changed, 0 insertions(+), 0 deletions(-)

 create mode 100644 train.py

 create mode 100644 utils.py

Create a repository on GitHub called `ml-project`

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git branch -M main

Link your local repository to this GitHub repository

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git remote add origin https://github.com/HarryAI-6748/ml-project.git

Push your commits to GitHub

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git push -u origin main

info: please complete authentication in your browser...

Enumerating objects: 3, done.

Counting objects: 100% (3/3), done.

Delta compression using up to 8 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 288 bytes | 288.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

To https://github.com/HarryAI-6748/ml-project.git

 * [new branch]   main -> main

branch 'main' set up to track 'origin/main'.

The commands to retrieve your teammate's changes from GitHub

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git pull

The commands to stage and commit your local changes

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git add train.py

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git commit -m "First Changes"

[main a0b1b3a] First Changes

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

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project> git status

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

  (use "git push" to publish your local commits)

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: utils.py

The commands to push your work to GitHub

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
git pull

Already up to date.

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
git add utils.py

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
git status

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

  (use "git push" to publish your local commits)

Changes to be committed:

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

        modified: utils.py

Untracked files:

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

        README.md

        predict.py

PS S:\Practice Python yourself\AI projects\Resources_Notes_datasheets\ml-project>
git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 8 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 359 bytes | 359.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

To https://github.com/HarryAI-6748/ml-project.git

   67bdd0b..a0b1b3a main -> main

What potential issues might arise and how to handle them

1.   The user might push changes without pulling the recent existing changes in remote repository.
2.   The user might push the changes into the wrong branch.
3.   The user might lose the entire project due to loss of physical hardware and not updating the remote repository first-hand.
4.   Users might   commit the changes directly without adding it to the staging area which may lead to unchecked errors.