1. Git Init

$ git init

Команда Git init використовується для створення нового порожнього сховища або повторної ініціалізації існуючого робочого проекту як сховища Git.

Після виконання команди $ git init підкаталог .git, який містить усі метадані, буде створено всередині робочого каталогу.

Тож після створення git repository, ви можете зв’язати його з віддаленим репозиторієм за допомогою команди:

$ git remote add remoteRepoURL

2. Git Add

Ця команда використовується для додавання контексту файлу до індексу. Команда git add оновлює вміст поточного робочого дерева до проміжної версії та готує його до наступного коміту.

Щоб додати один файл:

$ git add fileName

Щоб додати всі файли:

$ git add .

3. Git Commit

$ git commit

Команди Git commit для запису змін до репозиторію.

Щоб закріпити за допомогою повідомлення:

$ git commit -m “commitMessage”

Щоб змінити повідомлення коміту:

$ git amend

4. Git Clone

$ git clone repositoryURL

Ми використовуємо цю команду, щоб клонувати репозиторій у новий каталог.

Щоб клонувати репозиторій у певну локальну папку:

$ git clone

repositoryURL “ffolderPath”

Щоб клонувати гілку:

$ git clone -b branchName repositoryURL

5. Git Tag

Ця команда створює, перераховує, видаляє або перевіряє тег.

Щоб створити тег:

$ git tag tagName

Щоб отримати список тегів:

тег $ git

6 . Git Stash

$ git stash

Ви можете перемикати гілки, не закріплюючи поточну, за допомогою команди git stash.

Щоб сховати повідомлення:

$ git stash save stashMessage

Щоб отримати список схованок:

$ git stash list

Щоб повторно застосувати зміни, які були приховані:

$ git stash apply

інакше ви можете використати

$ git stash pop

Ключова відмінність між цими двома операціями полягає в тому, що stash pop видаляє схованку з стек після його застосування.

Щоб видалити останній сховок:

$ git stash drop

Щоб видалити всі доступні тайники:

$ git stash clear

The project was created by Linus Torvalds to manage the development of the Linux kernel,