



# **10 GIT Commands**

**EVERY DEVELOPER  
MUST KNOW**

# 1. Git Init

---

**Initializes a new Git repository in your project folder, creating a .git directory to track changes.**



# 2. Git Clone

---

**Copies a remote repository to your local machine, allowing you to work on it locally.**

A stylized illustration of a terminal window. The window has a dark gray title bar with a white circle on the left and two white square buttons with left and right arrow symbols on the right. The main area of the window is black with white text. The text shows a command prompt followed by the command to clone a repository.

```
>> git clone  
[repository_url]
```

# 3. Git Add

---

**Adds a specific file to the staging area, preparing it for the next commit.**

A stylized illustration of a terminal window with a dark blue background and a light blue border. The window has a title bar at the top with a red close button on the left and two yellow maximize/minimize buttons on the right. The terminal content is white text on a dark background, showing the command to add a file to the staging area.

```
>> git add  
[file_name]
```

# 4. Git Status

---

**Shows the current status of your repository, displaying modified, staged, and untracked files.**



# 5. Git Commit

---

**Records changes in the repository, accompanied by a brief commit message.**

A stylized illustration of a terminal window with a dark background and light gray text. The window has a title bar at the top with a small circle on the left and two square buttons with left and right arrow symbols on the right. The text inside the terminal shows a command being entered: a prompt followed by 'git commit -m' and a commit message in quotes on the next line.

```
>> git commit -m  
"commit_message"
```

# 6. Git Pull

---

**Fetches and merges changes from the remote repository to your local branch.**



# 7. Git Push

---

**Uploads your local commits to the remote repository, updating the shared codebase.**





# 8. Git Branch

---

**Lists all branches in the repository, highlighting the current branch.**



# 9. Git Checkout

---

**Switches to the specified branch, allowing you to work on that branch.**



```
>> git checkout  
[branch_name]
```

# 10. Git Merge

---

**Integrates changes from the specified branch into the current branch.**

A stylized illustration of a terminal window with a dark background and a light border. The window has a title bar at the top with a small circle on the left and two square buttons with left and right arrow symbols on the right. The main area of the window contains the text '>> git merge' followed by '[branch\_name]' on the next line, both in a light-colored monospace font.

```
>> git merge  
[branch_name]
```



---

**Sarthak Chauhan**

---

**FOUND THESE HELPFUL?**

Follow me for more such coding  
related content.