Git & GitHub Essential Commands Guide

1. Git Basics

- `git init` – Initialises a new Git repository in your project.
- `git clone <repo_URL>` – Creates a local copy of a remote repository.
- `git status` – Shows the current state of the working directory.
- `git add <file>` – Stages specific files for the next commit.
- `git add .` – Stages all changes.
- `git commit -m "message"` – Saves staged changes with a descriptive commit message.

2. Branching & Merging

- `git branch` – Lists all branches in the repository.
- `git branch <branch_name>` – Creates a new branch.
- `git checkout <branch_name>` – Switches to an existing branch.
- `git checkout -b <branch_name>` – Creates and switches to a new branch.
- `git merge <branch_name>` – Merges another branch into the current branch.
- `git branch -d <branch_name>` – Deletes a local branch.

3. Working with Remotes

- `git remote -v` – Lists connected remote repositories.
- `git remote add origin <repo_URL>` – Links local repository to a remote.
- `git push -u origin <branch_name>` – Pushes local commits to a remote repository.
- `git pull origin <branch_name>` – Fetches and updates local repository with remote changes.
- `git fetch` – Retrieves updates from a remote repository without merging them.

4. Undoing Changes

- `git reset --soft <commit_hash>` – Resets to a specific commit but keeps changes staged.

- `git reset --hard <commit_hash>` – Resets to a specific commit and deletes all changes.
- `git revert <commit_hash>` – Creates a new commit that undoes changes from a previous commit.
- `git checkout -- <file>` – Discards changes in a specific file.

5. Stashing Work

- `git stash` – Saves uncommitted changes temporarily.
- `git stash apply` – Restores the latest stashed changes.
- `git stash drop` – Deletes the latest stash entry.
- `git stash list` – Displays all stashed changes.

6. Collaboration on GitHub

- `git push origin main` – Pushes the local repository to GitHub.
- `git pull origin main` – Pulls updates from GitHub.
- `git fork <repo_URL>` – Copies another repository into your GitHub account.
- `git clone <your_forked_repo_URL>` – Clones a forked repository locally.
- `git fetch upstream` – Retrieves changes from the original repo.
- `git merge upstream/main` – Merges the latest changes from the original repo into your branch.

7. Viewing History

- `git log` – Displays a list of commit history.
- `git log --oneline` – Shows a brief history of commits.
- `git diff` – Compares changes between commits.
- `git blame <file>` – Shows who last modified each file line.

This should serve as a **comprehensive guide** for Git and GitHub commands crucial in project management and development workflows. You can copy and paste this into a document and then save it as a **PDF** using Microsoft Word or Google Docs. 😊