# Case Studies on Git Commands

## Cloning a Public Repository

Scenario: You need to clone a public GitHub repository to your local system.  
Solution:  
1. Identify the repository URL (e.g., https://github.com/user/repo.git).  
2. Run the command:  
 git clone https://github.com/user/repo.git

## Cloning a Private Repository

Scenario: You want to clone a private repository but face permission issues.  
Solution:  
1. Ensure you have access to the repository.  
2. Use SSH or Personal Access Token for authentication:  
 git clone git@github.com:user/repo.git

## Cloning a Specific Branch

Scenario: You only need to work on a specific branch of a repository.  
Solution:  
 git clone -b branch\_name https://github.com/user/repo.git

## Cloning with Depth

Scenario: You want to clone a repository with only the latest commit to save space.  
Solution:  
 git clone --depth 1 https://github.com/user/repo.git

## Push Changes to a New Repository

Scenario: You've initialized a local repository and need to push it to GitHub.  
Solution:  
1. Create a repository on GitHub.  
2. Add the remote URL:  
 git remote add origin https://github.com/user/repo.git  
3. Push the code:  
 git push -u origin main

## Push to an Existing Repository

Scenario: You’ve made local changes and want to push them to an existing repository.  
Solution:  
1. Commit changes:  
 git add .  
 git commit -m "Your message"  
2. Push changes:  
 git push

## Resolving Push Rejections

Scenario: Your push fails due to divergent branches.  
Solution:  
1. Fetch and merge remote changes:  
 git pull origin branch\_name  
2. Resolve conflicts and push again:  
 git push

## Force Push

Scenario: You want to overwrite remote changes (not recommended unless necessary).  
Solution:  
 git push --force

## Push a Tag

Scenario: You’ve created a tag and want to push it to the remote repository.  
Solution:  
 git push origin tag\_name

## Pull Changes

Scenario: Your local branch is behind the remote branch.  
Solution:  
 git pull origin branch\_name

## Pull and Rebase

Scenario: You want to rebase local commits on top of the latest remote changes.  
Solution:  
 git pull --rebase origin branch\_name

## Handling Pull Conflicts

Scenario: Pulling changes results in a conflict.  
Solution:  
1. Resolve conflicts manually.  
2. Add resolved files:  
 git add conflict\_file  
3. Continue the rebase or merge process:  
 git rebase --continue

## Clone Using SSH

Scenario: You prefer SSH over HTTPS for cloning.  
Solution:  
1. Set up SSH keys on GitHub.  
2. Clone the repository:  
 git clone git@github.com:user/repo.git

## Push to a Forked Repository

Scenario: You forked a repository and want to push changes to your fork.  
Solution:  
1. Add your fork as a remote:  
 git remote add fork https://github.com/your\_username/repo.git  
2. Push changes to your fork:  
 git push fork branch\_name

## Pull from Upstream

Scenario: Your fork is behind the original repository, and you want to sync it.  
Solution:  
1. Add the upstream repository:  
 git remote add upstream https://github.com/original/repo.git  
2. Fetch changes and merge:  
 git pull upstream main

## Pull with a Specific Depth

Scenario: You only need the latest few commits while pulling.  
Solution:  
 git pull --depth 5 origin branch\_name

## Push to a Different Branch

Scenario: You want to push changes to a branch other than the current one.  
Solution:  
 git push origin your\_branch\_name

## Clone a Submodule

Scenario: You need to clone a repository that includes submodules.  
Solution:  
 git clone --recurse-submodules https://github.com/user/repo.git

## Pull with Stash

Scenario: You have uncommitted changes and need to pull updates.  
Solution:  
1. Stash your changes:  
 git stash  
2. Pull updates:  
 git pull  
3. Apply your stashed changes:  
 git stash pop

## Push a New Branch

Scenario: You’ve created a new branch and want to push it to the remote repository.  
Solution:  
1. Create a branch:  
 git checkout -b new\_branch  
2. Push the branch:  
 git push -u origin new\_branch