1. git pull: This command is used to fetch and merge changes from a remote repository into your local repository. It is useful for keeping your local codebase up-to-date with the remote version.
2. git tag: This command is used to create, list, delete, or annotate tags on Git commits. Tags are useful for marking specific points in your project's history, such as release versions.
3. git push --tags: This command is used to push your local tags to the remote repository. This is necessary if you want your tags to be available to other collaborators.
4. git release: While there is no built-in git release command, you can create your own release management workflow using a combination of Git commands like git tag, git push, and potentially external tools or scripts.
5. git ignore: This command is not directly available, but you can manage your .gitignore file, which specifies which files and directories should be ignored by Git. This is useful for excluding temporary files, compiled artifacts, and other untracked content.
6. ssh-keygen: This command is used to generate SSH keys, which can be used for secure authentication when interacting with remote Git repositories over the SSH protocol.
7. git config --global user.name "Your Name": This command sets your global Git username, which is useful for properly attributing your commits.
8. git config --global user.email "your.email@example.com": This command sets your global Git email address, which is also important for properly attributing your commits.
9. git remote add origin git@github.com:your-username/your-repository.git: This command is used to add a remote Git repository, which is necessary for pushing and pulling changes.
10. git push -u origin master: This command is used to push your local "master" branch to the remote "origin" repository, and sets the upstream branch so that future git push commands can omit the remote and branch names.