

GIT COMMAND CHEAT SHEET (`nsaw/` edition)

Your local repo: `/Users/sawyer/gitSync`

Remote origin: `https://github.com/nsaw/``

Basic Setup and Configuration

`git init`

Initialize a repo in current dir

`git clone https://github.com/nsaw/``

Clone an existing repo

`git config`

`--global user.name "nsaw"`

`git config`

`--global user.email "nick@sawyerdesign.io"`

Set up your identity globally

Working with Files

`git add <file>`

Stage file for commit

`git commit -m "Message"`

Commit staged changes

`git status`

See what's going on

`git diff`

See what's changed

(unstaged vs working dir)

`git diff --staged`

See changes staged for commit

`git reset <file> #`

Unstage a file

`git restore <file>`

Restore file to

last committed version

`git revert <commit_hash>`

Create new commit that undoes

previous commit (safe)

`git rm <file>`

Delete file from disk and staging

Pro Moves (Recommended)

```

`git cherry-pick <commit>`

# Apply specific commit from another branch

`git reflog`

# Git's black box: everything you've done

`git bisect start`

# Binary search for bugs in history

`git reset --hard <commit>`

# Go back in time (destructive)

`git switch -`

# Go back to previous branch

## ## Sawyer GitSync Flow (Typical)

`cd /Users/sawyer/gitSync`

`git pull origin main`

# Always pull latest

`git checkout -b feature/my-thing`

# Branch off

# ...code...

`git add .`

`git commit -m "Add feature"`

`git push origin feature/my-thing`

# Push feature for review

`git checkout main`

`git pull origin main`

`git merge feature/my-thing`

`git push origin main`

# Merge & deploy

## ## Branch Management

```
git branch
List all branches
```

```
git branch <name>
Create new branch
```

```
git checkout <name>
Switch to branch
```

```
git switch <name>
Also switches branch (modern alternative)
```

```
git checkout -b <name>
Create AND switch to new branch
```

```
git merge <name>
Merge into current branch
```

```
git rebase <name>
Rebase onto another branch (rewrite history)
```

```
git branch -d <name>
Delete branch (safe)
```

```
git branch -D <name>
Force delete branch (danger zone)
```

## ## Remote Repos

```
git remote add origin https://github.com/nsaw/# Connect to
remote repo
```

```
git push origin <branch>
Push local commits to remote
```

```
git pull origin <branch>
Pull & merge from remote
```

```
git fetch origin
Pull changes, don't merge
```

```
git remote -v
Show remotes
```

## ## History & Logs

```
git log
View commit history
```

```
git log --oneline --graph --all # Compact,
visual history
```

```
git blame <file>
See who wrote which line
```

```
git show <commit>
Show what a commit changed
```

## ## Cleanup & Stashing

```
git stash
Save dirty changes temporarily
```

```
git stash pop
Reapply last stash
```

```
git stash list
List all stashes
```

```
git stash drop
Delete a stash
```

```
git clean -fd
Delete untracked files & dirs
```

## ## Tags & Checkpoints

```
git tag <name>
Tag current commit
```

```
git tag -a <name> -m "Message" # Annotated
tag
```

```
git push origin <tag>
Push a tag
```

```
git push origin --tags
Push all tags
```