

# Git Cheat Sheet

## 1 Configuration

```
# Set username and email
git config --global user.name "Your Name"
git config --global user.email "your.email@example.com"

# Check current configuration
git config --list

# Set default editor
git config --global core.editor "vim"

# Enable color output
git config --global color.ui auto
```

## 2 Initializing and Cloning

```
# Initialize a new Git repository
git init

# Clone a repository
git clone <repo-url>

# Clone a repository with a different name
git clone <repo-url> <new-folder-name>
```

## 3 Basic Commands

```
# Check status of repository
git status

# Add all changes to staging area
git add .

# Commit changes with message
git commit -m "Your commit message"
```

## 4 Branching

```
# List branches
git branch

# Create and switch to a new branch
git checkout -b <branch-name>

# Delete a branch
git branch -d <branch-name>
```

## 5 Merging and Rebasing

```
# Merge a branch into current branch
git merge <branch-name>

# Rebase onto another branch
git rebase <branch-name>
```

## 6 Remote Repositories

```
# Add a remote repository
git remote add origin <repo-url>

# Push changes to remote repository
git push origin <branch-name>
```

## 7 Undoing Changes

```
# Revert to a previous commit
git revert <commit-hash>

# Reset to a previous commit (hard reset)
git reset --hard <commit-hash>
```

## 8 Stashing Changes

```
# Stash current changes
git stash

# Apply the most recent stash
git stash pop
```

## 9 Viewing Logs and History

```
# Show commit history
git log

# Show commit history in one line
git log --oneline
```

## 10 Tagging

```
# Create a lightweight tag
git tag <tag-name>

# Push a tag to remote
git push origin <tag-name>
```

## 11 Working with Submodules

```
# Add a submodule
git submodule add <repo-url>
```

## 12 Git Ignore

Create a '.gitignore' file to ignore specific files:

```
# Ignore all '.log' files
*.log

# Ignore a specific folder
/node_modules/
```

## 13 Git Aliases

```
# Create an alias for a command
git config --global alias.st status
```

## 14 Security & Authentication

```
# Generate SSH key
git ssh-keygen -t rsa -b 4096 -C "your.email@example.com"
```

## 15 Force Push & Fix Mistakes

```
# Force push changes (use with caution)
git push --force
```