

Git Commands Cheat Sheet for Product Development

Git Configuration

```
git config --global user.name "Your Name"
git config --global user.email "your@email.com"
```

Starting a Repository

```
git init                # Start a new repo
git clone <repo-url>    # Clone from remote
```

Working with Files

```
git add file.txt        # Add single file
git add .               # Add all modified/new files
```

- Unstaged: After editing, before git add
- Staged: After git add, before git commit

Committing Changes

```
git commit -m "type: message"
```

Commit Types:

- feat: Add new feature
- fix: Bug fix
- docs: Documentation change
- style: Code formatting only
- refactor: Code change without new feature/bug
- test: Add/update tests
- chore: Maintenance (build, deps, etc.)

Examples:

```
git commit -m "feat: add login feature"
git commit -m "fix: resolve crash in checkout flow"
```

Synchronizing Code

```
git pull origin branch-name    # Pull latest changes
```

Git Commands Cheat Sheet for Product Development

`git push origin branch-name` # Push local commits

- After git commit, use git push to update remote repo.

Branching

`git branch` # List local branches

`git checkout -b feature/new-ui` # Create and switch to a new branch

`git checkout main` # Switch to another branch

`git branch -d feature/new-ui` # Delete local branch

Merging

`git checkout develop` # Go to the target branch

`git merge feature/new-ui` # Merge feature branch into it

Handling Conflicts

- Resolve in code editor

Then run:

`git add .`

`git commit -m "fix: resolve merge conflict"`

Versioning & Tags

`git tag -a v1.0 -m "Release v1.0"`

`git push origin v1.0`

Cleaning Up

`git branch -d branch-name` # Delete local

`git push origin --delete branch-name` # Delete remote

Best Practice Tips

- Keep branches focused (feature/payment, bugfix/login-error)

- Always pull before pushing

Git Commands Cheat Sheet for Product Development

- Use meaningful commit messages
- Regularly merge develop to keep features updated
- Create pull requests for merging into main