

Essential Git & GitHub Workflow for Beginners

Subtitle: Learn Git version control and GitHub workflow to manage projects, collaborate, and avoid common errors.

Website Name: haas.dev

Website Link: <https://dev-roast-app.vercel.app>

Introduction

Git and GitHub are **essential tools for every developer**. They help you track code changes, collaborate with others, and deploy projects efficiently. This guide teaches beginners the **fundamentals of Git, GitHub workflow, and practical best practices**.

Step 1: Installing Git & Setting Up

- **Install Git:** <https://git-scm.com/downloads>
- **Configure Git:**

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

- **Verify installation:**

```
git --version
```

Step 2: Initialize a Local Repository

```
mkdir my-project  
cd my-project  
git init
```

- Creates a `.git` folder to track changes
-

Step 3: Basic Git Commands

1. **Check status:**

```
git status
```

2. Add files to staging:

```
git add index.html
```

- Add all files: `git add .`

3. Commit changes:

```
git commit -m "Initial commit"
```

4. View commit history:

```
git log
```

Step 4: Branching & Merging

- Create a new branch:

```
git checkout -b feature-branch
```

- Switch branches:

```
git checkout main
```

- Merge branch:

```
git merge feature-branch
```

Step 5: Connect to GitHub

1. Create a repository on GitHub
2. Add remote to local repo:

```
git remote add origin https://github.com/username/repo.git
```

3. Push code:
-

```
git push -u origin main
```

Step 6: Pull & Fetch Updates

- **Fetch changes from remote:**

```
git fetch
```

- **Pull latest changes:**

```
git pull origin main
```

Step 7: Collaboration & Pull Requests

- Fork a repository on GitHub
- Clone it locally:

```
git clone https://github.com/user/repo.git
```

- Make changes → commit → push → **create a pull request** on GitHub
-

Step 8: Handling Common Issues

- **Merge conflicts:** Resolve manually, then commit
- **Forgot to add files before commit:**

```
git add <file>  
git commit --amend
```

- **Undo local changes:**

```
git checkout -- <file>
```

Step 9: Mini Project Idea

- Create a personal project folder
- Initialize Git, make 5 commits with meaningful messages

- Push to GitHub
 - Create a new branch, make changes, merge into main
 - Open a pull request on a test repo
-

Step 10: Key Takeaways

- Git tracks changes and allows safe collaboration
 - GitHub makes it easy to host and share projects
 - Branching + pull requests are essential for teamwork
 - Practice with mini-projects to internalize commands
-

Visit **haas.dev** for step-by-step Git & GitHub guides, beginner-friendly workflows, and project tutorials.

Website Name: haas.dev

Website Link: <https://dev-roast-app.vercel.app>
