## 1️⃣ Create a Local & Remote Repository and Merge Them

### 📌 Step 1: Create a Local Repository

mkdir my\_project  
cd my\_project  
git init

* Creates a new project folder and initializes Git.

### 📌 Step 2: Add a File & Commit

echo "# My Project" > README.md  
git add README.md  
git commit -m "Initial commit"

* Creates a README.md file, stages it, and commits it.

### 📌 Step 3: Create a Remote Repository on GitHub/GitLab

1. Go to GitHub/GitLab.
2. Create a new repository **(without initializing README or .gitignore)**.
3. Copy the remote repository URL.

### 📌 Step 4: Link Local Repo to Remote

git remote add origin <REMOTE\_URL>  
git branch -M main  
git push -u origin main

* Links the local repo to the remote repository (origin).
* Pushes the main branch to the remote repository.

✅ **Now your local and remote repos are synchronized!**

## 2️⃣ Create a Child Branch, Update Code & Push It

### 📌 Step 1: Create a New Branch

git checkout -b feature-branch

* Creates and switches to feature-branch.

### 📌 Step 2: Update Code & Commit

echo "New feature added" >> feature.txt  
git add feature.txt  
git commit -m "Added new feature"

* Adds a new file and commits the changes.

### 📌 Step 3: Push the Branch to Remote

git push -u origin feature-branch

* Pushes feature-branch to the remote repository.

✅ **Your feature branch is now on GitHub/GitLab!**

## 3️⃣ Raise a Pull Request & Merge It

### 📌 Step 1: Open a Pull Request (PR)

1. Go to your **GitHub/GitLab repository**.
2. Click **Compare & pull request** (GitHub) or **New Merge Request** (GitLab).
3. Select feature-branch → main branch.
4. Add a title & description, then **Submit PR**.

### 📌 Step 2: Review & Merge PR

1. The team reviews the changes.
2. Click **Merge Pull Request** (GitHub) or **Merge** (GitLab).
3. Choose **Squash, Merge, or Rebase** (based on strategy).

### 📌 Step 3: Delete Merged Branch

git branch -d feature-branch # Delete locally  
git push origin --delete feature-branch # Delete remotely

* Cleans up the feature branch after merging.

### 📌 Step 4: Update Local main Branch

git checkout main  
git pull origin main

* Ensures the local main branch is up to date.

✅ **Now the feature is merged into main successfully!** 🚀

## Summary of Commands

| **Step** | **Command** |
| --- | --- |
| **Initialize Local Repo** | git init |
| **Create & Push First Commit** | git add . → git commit -m "message" → git push -u origin main |
| **Create Feature Branch** | git checkout -b feature-branch |
| **Make Changes & Push** | git add . → git commit -m "message" → git push -u origin feature-branch |
| **Raise a Pull Request** | Via GitHub/GitLab UI |
| **Merge & Delete Branch** | git branch -d feature-branch → git push origin --delete feature-branch |
| **Update Local main** | git checkout main → git pull origin main |