

## GIT PRACTICE INDIVIDUAL

### 1. Create and Clone a GitHub Repository

1. Go to [github.com](https://github.com) and create a new repository (e.g., git-practice).
2. Initialize with README or .gitignore.
3. Clone your repository locally using your terminal.

```
git clone <repository-address>
```

4. Navigate into the repository the clone command has just created

```
cd git-practice
```

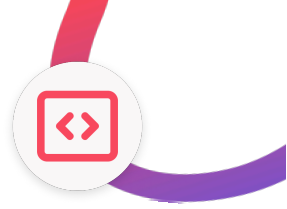
### 2. Create a dev Branch and Push It Upstream

1. Create the new local branch name **dev** from **main** and switch to it.

```
git checkout -b dev
```

2. Create a file named index.html and add the following content:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Git practice</title>
```



```
</head>
<body>

</body>
</html>
```

3. Add the changes to the stage.

```
git add .
```

4. Commit the changes to store them in the git history.

```
git commit -m "<commit-message>"
```

5. Push the local **dev** branch into the remote.

```
git push -u origin dev
```

### 3. Create 3 Feature Branches From dev

1. Check if you are in the **dev** branch.

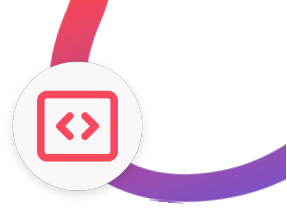
```
git branch
```

2. Switch to the dev branch (omit this step if you are already on the Dev branch).

```
git checkout dev
```

3. Create three branches with the names **feature/header**, **feature/main** and **feature/footer**.

```
git branch feature/header
git branch feature/main
git branch feature/footer
```



4. **Push the three branches you have created upstream** (make sure you are in the branch you want to push into the remote when doing the push command).

```
git checkout feature/header  
git push -u origin feature/header
```

```
git checkout feature/main  
git push -u origin feature/main
```

```
git checkout feature/footer  
git push -u origin feature/footer
```

#### 4. Make changes in the feature branches and save them in history

1. Check the branch you are in

```
git branch
```

2. Switch to **feature/header** if you are not in it.

```
git checkout feature/header
```

3. Add the following content inside the **<body>** tag in the index.html file:

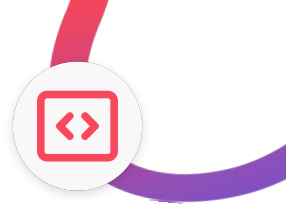
```
<header>I am a header</header>
```

4. Add and commit the changes to store them in the git history of the branch.

```
git add .  
git commit -m <commit-message>
```

5. Push the updated content into the remote

```
git push
```



6. Repeat steps 1 to 5 for the feature/main and feature/footer branches, but adding in step 3 the following content instead:

**In feature/main:**

```
<main>I am main</main>
```

**In feature/footer:**

```
<footer>I am a footer</footer>
```

## 5. Create a Pull Request for the three feature branches to merge into dev

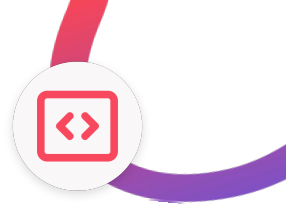
1. Go to your repository in [GitHub](#).
2. Click on the Pull Request link on the navigation bar at the top of the repo.



3. Click on the action button “New Pull Request”.
4. Create a pull request from **feature/header** → **dev**.
  - a. Set the base to **dev**
  - b. Set the compare to **feature/header**.
  - c. Add a description of what the code of the branch the PR is merging into dev is doing.
  - d. Click on the “Create Pull Request” button.

Here is a link to help you with that:

<https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/creating-a-pull-request>



5. Repeat steps 1 to 4 for **feature/main** and **feature/footer**.

## 6. Merge your first Pull Request Into dev

1. On GitHub, go to the Pull Request from **feature/header** and click **Merge pull request** → **Confirm Merge**.
2. Delete **feature/header** on GitHub by clicking the “Delete branch” button after confirming the merge.
3. Delete **feature/header** on your local repository.

```
git branch -d feature/header
```

## 7. Update Local dev and Remaining Feature Branches

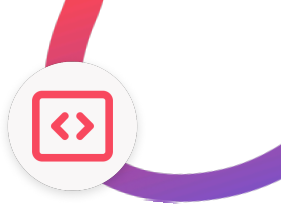
1. Update local **dev**.

```
git checkout dev  
git pull
```

2. Update local **feature/main**.

```
git checkout feature/main  
git pull origin dev
```

3. Solve merge conflict on **feature/main** in the merge editor:
  - a. Accept the Incoming Changes
  - b. Accept the Current Changes
  - c. Click on the solve merge button.



- d. Commit the changes -> `git commit -m "Merged dev into feature/main"`
- e. Push changes in remote -> `git push`

4. Repeat all of the steps in part 6. Merge your first Pull Request Into dev to merge feature/main into dev through the pull request.

5. Repeat all of the steps in 7. Update Local dev and Remaining Feature Branches and 6. Merge your first Pull Request Into dev to update and merge feature/footer.

## 8. Merge dev with main through a pull request

1. Create a Pull Request (PR) for dev.
  - a. Base should be set to main.
  - b. Compare should be set to dev.
  - c. Add a title
  - d. Add a description
  - e. Click the create pull request button.

2. Merge Pull Request.