

# the Master Course

{C0DENATION}

# React Intermediate

## GitHub Collaboration

{ CODENATION }

# Learning Objectives

**To explore the benefits of using branches in GitHub**

**To recognise common workflows for collaborating on projects**

**To recognise common workflows for contributing to projects**

**To identify merge conflicts after pulling code from a remote repository**

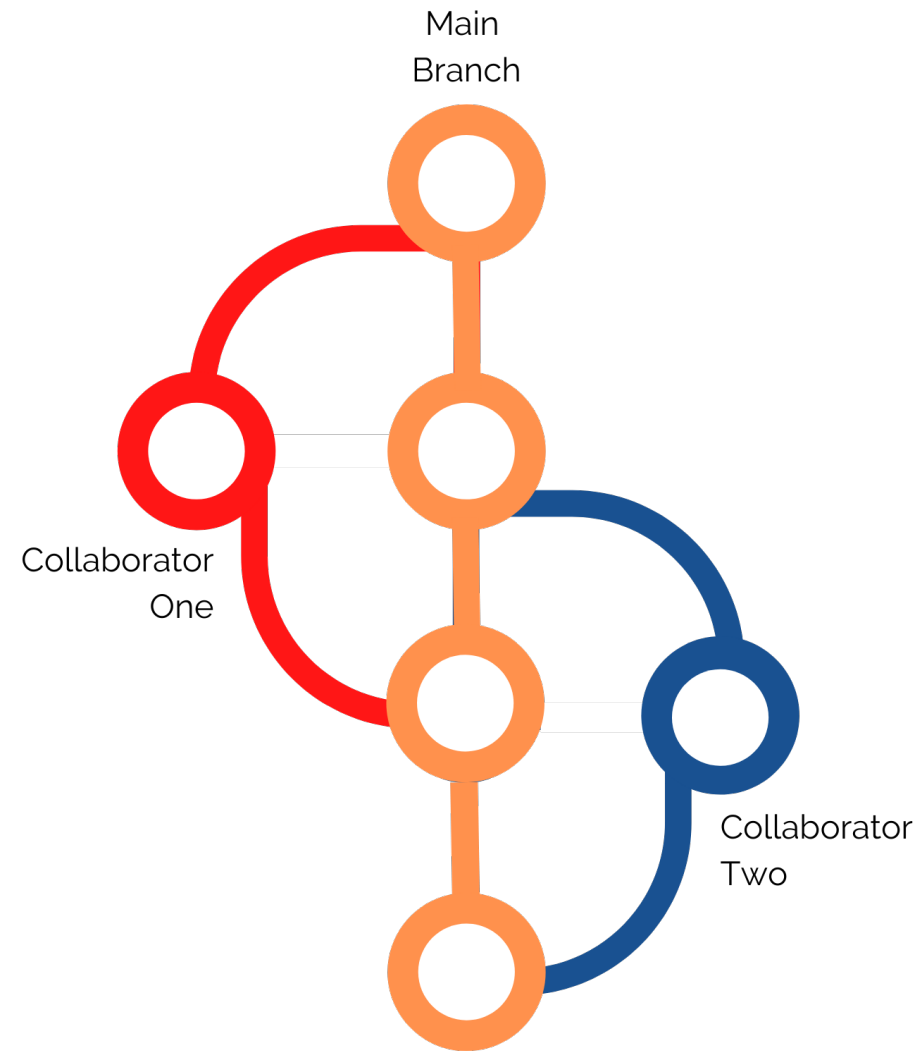


React

# What is Branching?

A common practice to isolate changes  
from the main branch in GitHub!

# React





React

# Why create a Branch?

Minimises the risk of breaking the website!

# Common Branch Commands

React

Displays the list of branches

- **git branch**

Creates a new branch and switches to that branch

- **git checkout -b branch name**

Deletes a branch

- **git branch -d branch name**

Moves to another branch

- **git checkout branch name**

# Common Git Commands

React

Adds all files to be staged to commit

- `git add .`

Creates a commit

- `git commit -m "your message"`

Pushes to remote repository

- `git push`

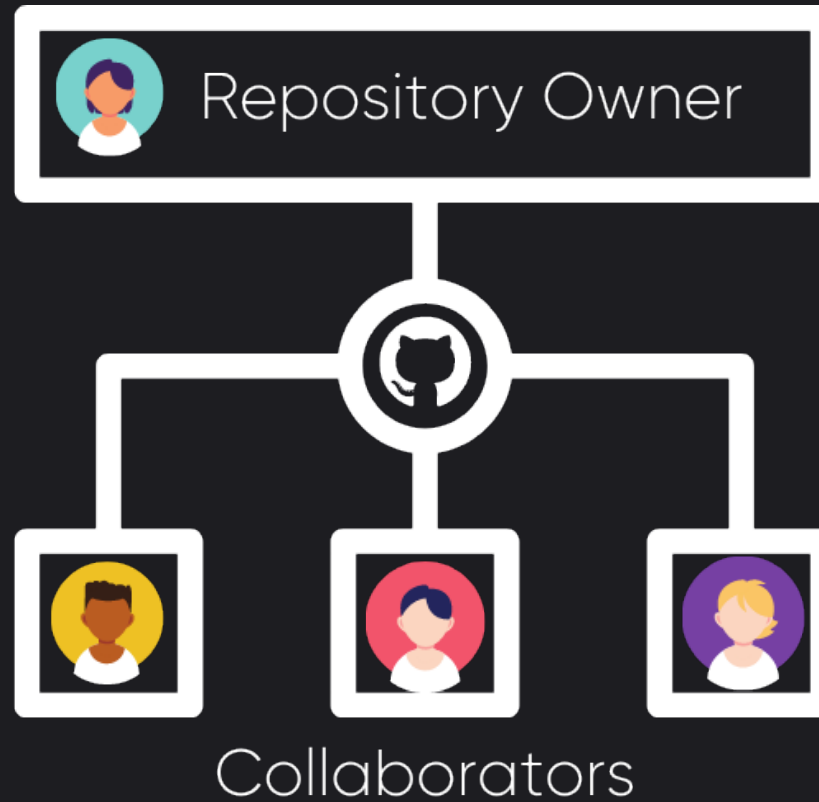
Pulls changes from remote repository

- `git pull`





# GitHub Collaboration



# Adding Collaborators

# React

## Adding Collaborators to GitHub

In your repository click settings, then select collaborators

The screenshot shows the GitHub repository settings page for a repository named 'git-demo'. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The Settings link is highlighted with an orange border. On the left sidebar, the 'Collaborators' option is selected and highlighted with a blue border. The main content area displays a message: 'You haven't invited any collaborators yet'. Below this message is a green 'Add people' button. To the right of the message is a link that says 'Click Add People'. Below the 'Add people' button is a section titled 'Add a collaborator to git-demo'. This section contains a search bar with the text 'Codenationers' and a green button that says 'Add Codenationers to this repository'. The search bar also shows a close button (X) and a dropdown menu with the text 'Codenationers'.

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Options Collaborators Security & analysis Branches Webhooks

You haven't invited any collaborators yet

Click Add People

Add people

Add a collaborator to **git-demo**

Add the GitHub usernames of the people you want to add to the repository.

Codenationers Codenationers

Add Codenationers to this repository



React

# Git Clone Repository

... Collaborators can now clone the repository


# Pull Requests

... create a pull request when a branch has been pushed.




# Pull Requests

# React


Back to GitHub and click the compare & pull request

 **feature/git-demo** had recent pushes less than a minute ago






Compare & pull request

 **main** ▾  **2 branches**  **0 tags**

Go to file Add file ▾ Code ▾

 **Codonationers** first commit

fe06469 7 hours ago ⌚ 2 commits

 .gitignore	first commit	7 hours ago
 README.md	first commit	7 hours ago
 app.js	first commit	7 hours ago
 index.html	first commit	7 hours ago
 styles.css	first commit	7 hours ago

# Pull Requests

# React

**If there are no merge conflicts, click create pull request.**

The screenshot displays the GitHub pull request creation interface. At the top, the 'base' is set to 'main' and the 'compare' branch is 'feature/git-demo'. A green checkmark and the text 'Able to merge.' are highlighted with an orange box, indicating that the branches can be automatically merged. Below this, the 'feature' branch is selected. The interface includes a 'Write' tab and a 'Preview' tab. A large text area for comments is present, with the placeholder text 'You can also leave an optional comment here.' At the bottom right, a green button labeled 'Create pull request' is highlighted with a blue box.

base: main ← compare: feature/git-demo ✓ Able to merge. These branches can be automatically merged.

feature

Write Preview H B I ≡ <> 🔗 ≡ ≡ ☑ @ ↗ ↶

Leave a comment

You can also leave an optional comment here.

Attach files by dragging & dropping, selecting or pasting them. M+


Create pull request ▼


# Pull Requests

# React

## Confirm merge branch to master branch

Add more commits by pushing to the **feature/git-demo** branch on **codenationers/git-demo**.






**This branch has no conflicts with the base branch**  
Merging can be performed automatically.

**Merge pull request** ▼

You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Add more commits by pushing to the **feature/git-demo** branch on **codenationers/git-demo**.



Merge pull request #1 from **codenationers/feature/git-demo**

feature

**Confirm merge**

Cancel



React

# What about merge conflicts?

You will encounter merge conflicts! No fear!



## What is a merge conflict?

Working on the same file will cause a merge conflict when pushing to Git.





# Merge Conflicts


React

## Resolving Merge Conflicts

Sometimes you will get this message on a pull request, click view pull request.

 base: main ← compare: conflict ✖ **Can't automatically merge.** Don't worry, you can still create the pull request.

 **conflict #2**  
No description available


 View pull request

You should attempt to resolve this conflict, click resolve conflicts.

Add more commits by pushing to the **conflict** branch on **codenationers/git-demo**



 **This branch has conflicts that must be resolved**  
Use the [web editor](#) or the [command line](#) to resolve conflicts.  
**Conflicting files**  
index.html

 Resolve conflicts

# Merge Conflicts

# React

## Resolving Merge Conflicts

Delete the lines you do not want to keep, click mark as resolved, then click commit merge.

```
index.html 1 conflict Prev Next ⚙ Mark as resolved

1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7     <title>Git Demo</title>
8   </head>
9   <body>
10    <h1>Git Demo</h1>
11    <p>git demo</p>
12 <<<<<< conflict
13 <p>I have created a conflict</p>
14 =====
15 <p>Testing a merge conflict</p>
16 >>>>>> main
17   </body>
18 </html>
19
```

Merge pull request and confirm merge.

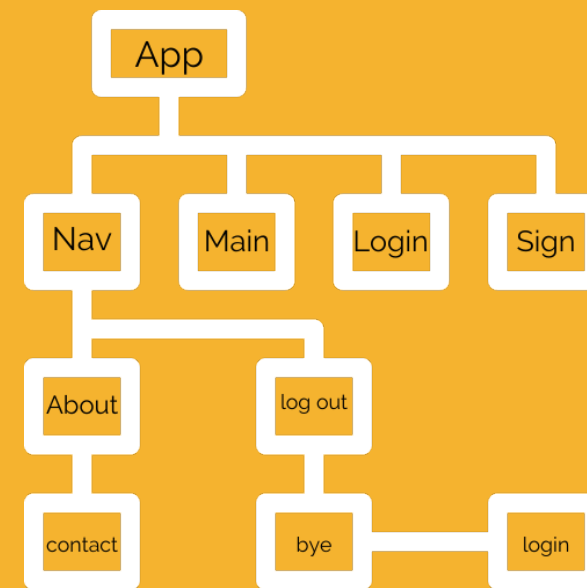


React

**How to avoid merge conflicts?**

# Planning!

# React





# Communication!

Regular stand ups

React





# Git Fork

A fork is a copy of a repository that allows you to freely experiment with changes without affecting the original project.

The owner of the original repository may choose to do a pull request on a forked repository.

React





# React

## Find out more

[Fork a repository](#)  
[Managing Branches](#)  
[Agile Trello Workflow](#)



# Learning Objectives

**To explore the benefits of using branches in GitHub**

**To recognise common workflows for collaborating on projects**

**To recognise common workflows for contributing to projects**

**To identify merge conflicts after pulling code from a remote repository**

## Activity

Create branches in your personal projects, name them appropriately and carry out pull requests in GitHub.