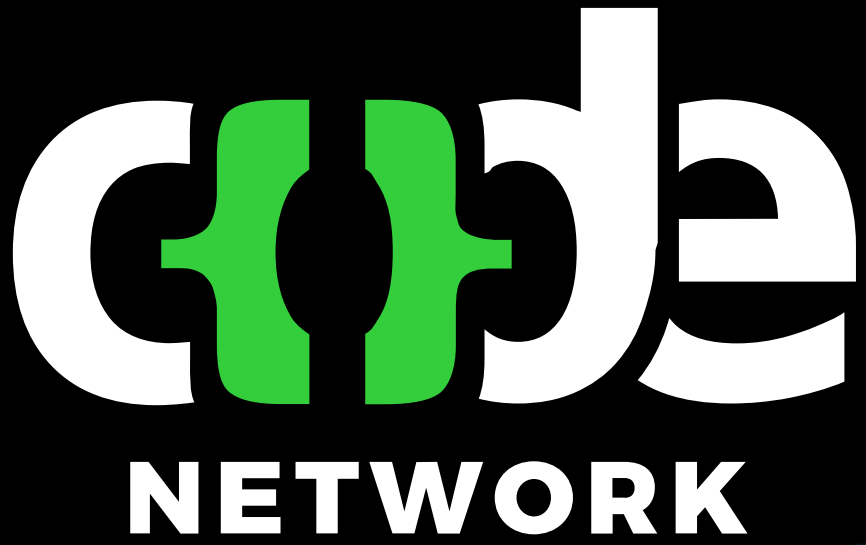
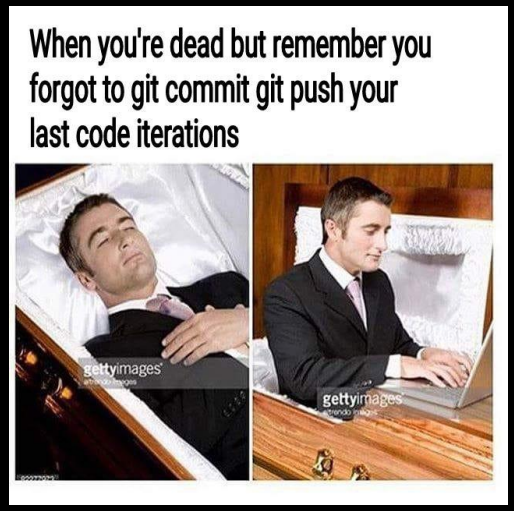
GIT 101

By Angus Wong
Code Network 2026



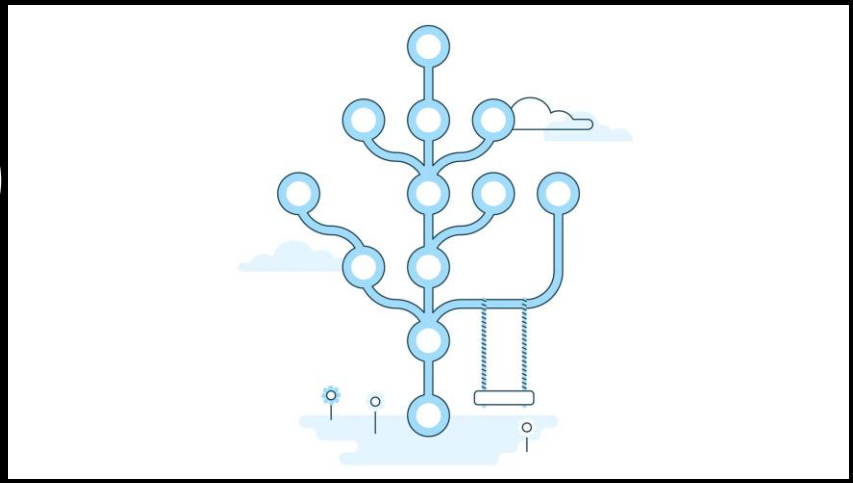
AGENDA

- Introduction to Git
- Git vs GitHub/GitLab/BitBucket
- Common Git Operations
- How to collaborate on Git
- Hands on execise



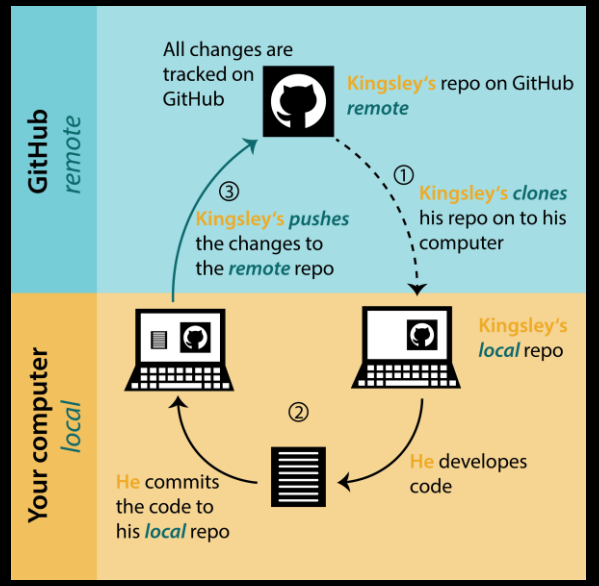
WHAT IS GIT

- Version control software tools have existed since 1960s
- Git is one of the most popular version control systems in the world
- Developed by Linus Torvalds (yes, that guy) in 2005 as an open-source version control software to be used on Linux
- Saves snapshot of all files inside a repository (folder)
- Allows programmers to view history and checkout the files' contents from any saved commits (version)



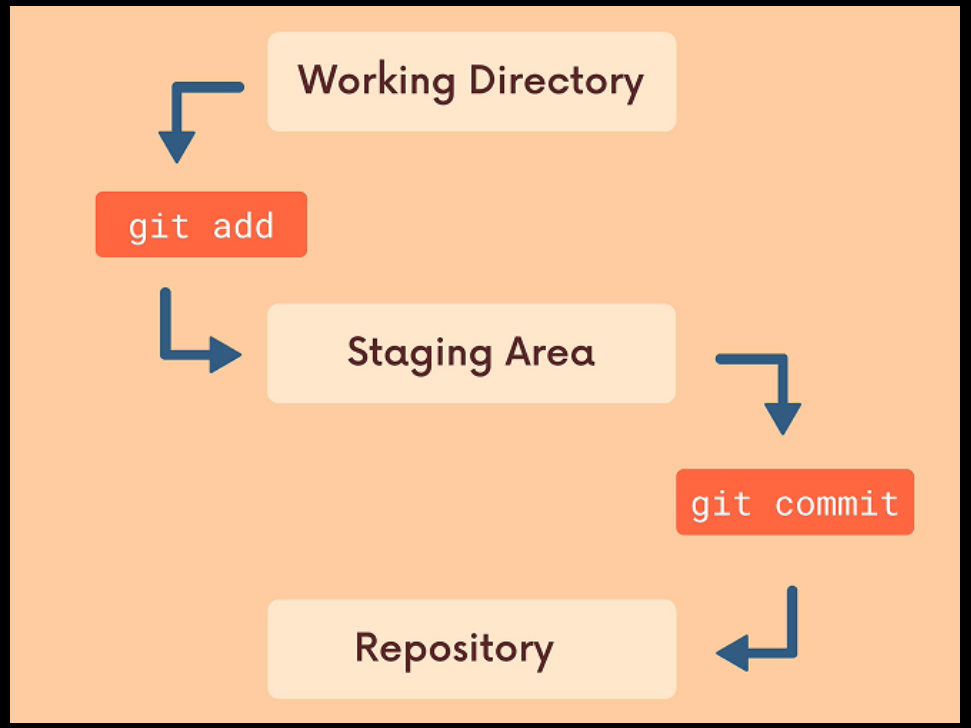
GIT VS GITHUB/GITLAB/BITBUCKET

- Git is a version control software. You can install Git on your computer
- GitHub/GitLab/BitBucket are collaborative version control platforms that use Git for distributed version control
- Git = File Explorer, GitHub = OneDrive/SharePoint
- You can download an entire folder (repo) from a remote (git clone)
- You can save the file history on your local Git, then upload them (git push) to the remote
- You can also get the remote history only (git fetch), or download the latest file changes from the remote (git pull)



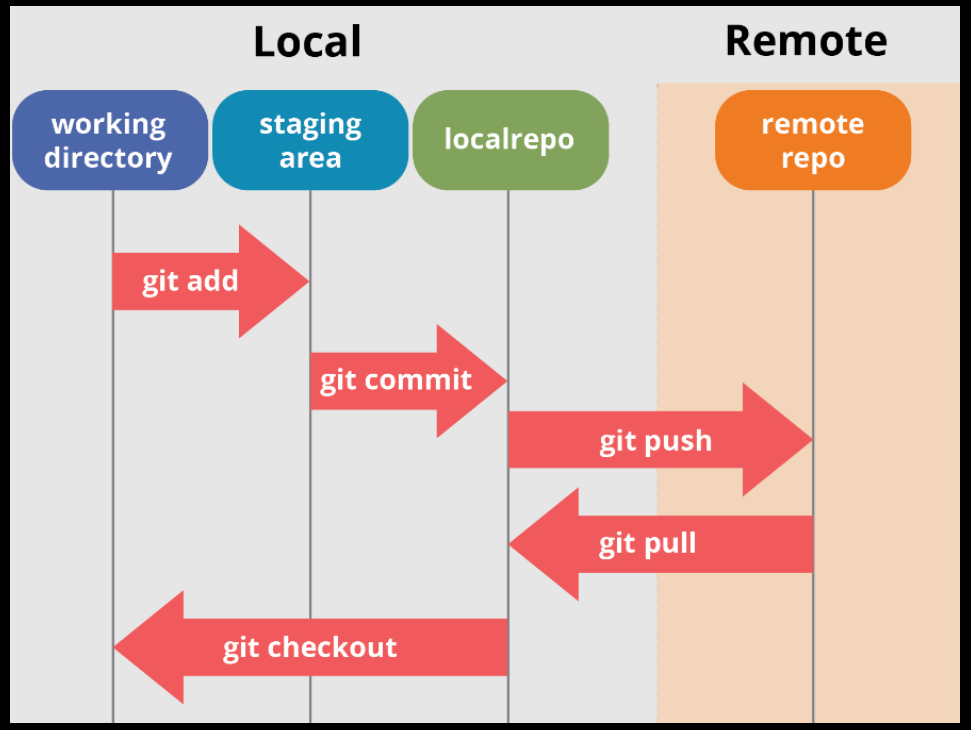
LOCAL GIT OPERATIONS

- Working Directory: Your local folder
- Staging Area: Contains file changes to be saved (committed)
- Repository: Saved change history
- git add <filename>: Tells git we want to save these changes
- git commit: Tells git to save the changes from the staging area
- git checkout <commit-id>: Look at the working directory's state from a commit



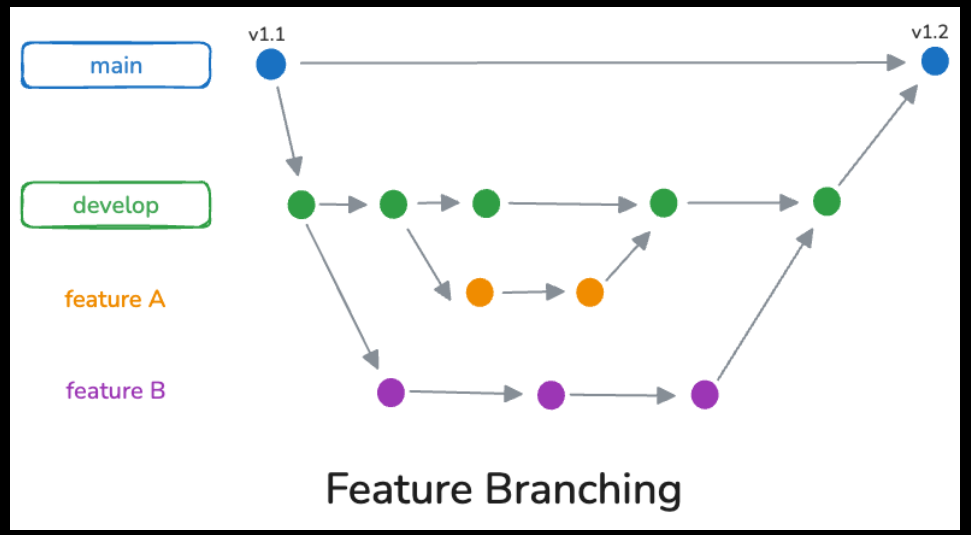
REMOTE GIT OPERATIONS

- git clone <repo-url>: Downloads remote repo (including working directory) to your computer
- git push: Uploads commits that only exist on your local git to the remote
- git fetch: Updates your local repo with the history from the remote (working directory unchanged)
- git pull: Does a git fetch, then git checkout the latest commit from remote (working directory will change)



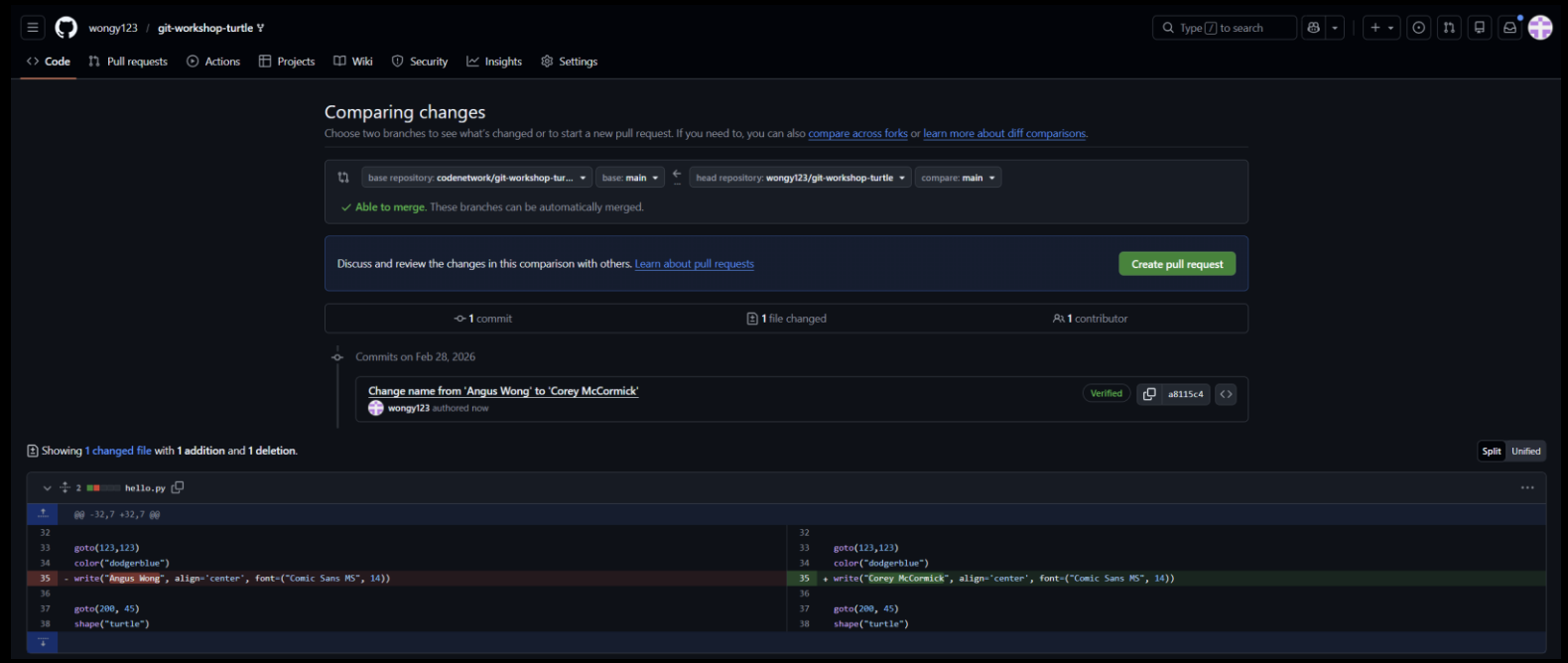
BRANCHING

- Branching allows for multiple developers to work on the same repo without interfering with each other's work
- Branches can be merged which will combine their changes and histories
- On remote Git platforms, you can create a Pull Request (PR) which is a request to merge a branch into another
- git branch <branch-name>: Creates branch
- git checkout –b <branch-name>: Checks out your working directory to that branch
- git merge <source-branch-name>: Merges the changes and history of the source branch into your current branch



FORKING

- Forking is available on remote Git platforms
- Forking creates a copy of the original repo under your control
- Make changes and push to your fork, then create a Pull Request to the original repo



Fork vs Branch

Comparison Chart

Fork	Branch
A fork is nothing but a copy of the repository to your own scope without affecting the original project repo.	A branch is an isolated environment to add, modify or delete a portion of the code without messing with the main code base.
Forking is a cloning operation in Git that is executed on the entire repository level.	Branching is a cloning operation in Git executed on a single repository.
Forking creates a full copy of the original repository which sits in your account.	Branching creates a branch to implementing your changes without affecting other developers.
The purpose is to improve someone else's project by adding some new features to the existing repo.	The purpose is to divert from the original code base without affecting other developers' work.

DB Difference Between.net

WORKSHOP

- Navigate to https://github.com/codenetwork/git-101-2026
- Read the instructions and set up your Git
- Fork the repo, then clone your own forked repo
- Add your name to Program.cs, add to stage, commit, and push
- Create a pull request to the original repo
- Put your hand up and a facilitator will help merge it!
- See the results: https://codenetwork.github.io/git-101-2026/

THANK YOU FOR JOINING US!

We hope you enjoy collaborating on our projects!