CI/CD Pipelines

Practical Explanation with Github Actions/Workflows

Definitions

CI = Continuous Integration

Making sure the code always ***builds*** and ***passes all tests***

CD = Continuous Delivery

Making sure any code that is ***integrable*** is ***released***

CD = Continuous Deployment

Making sure any code that is ***integrable*** is ***deployed to production***

How are they applied?

- Usually hand-in-hand with Git
- You trigger them to run on certain branches, tags, pull requests, or manually

How to start utilizing them?

- Github Workflows/Actions ← We're doing this today
- Gitlab Pipelines
- CircleCI
- Jenkins
- Azure Pipelines
- …

They all use yaml format with slight differences in schemas

How do they work?

- They run on Virtual Machines (Linux, Windows, Mac)
- They clone the repo and checkout the branch/tag
- Run a group of commands/scripts in certain order
  - Bash (linux)
  - Powershell (windows
- They have permissions; by default they might be able to write to your repo, so be mindful of that when you use them