# CI/CD

Continuous Integration, Continuous Delivery

# What is CI/CD?

Continuous Integration: is a practice that encourages developers to integrate their code into a main branch of a shared repository early and often. Instead of building out features in isolation and integrating them at the end of a development cycle, code is integrated with the shared repository by each developer multiple times throughout the day.

Continuous Delivery: is an extension of continuous integration. It focuses on automating the software delivery process so that teams can easily and confidently deploy their code to production at any time. By ensuring that the codebase is always in a deployable state, releasing software becomes an unremarkable event, without any complicated rituals. Teams can be confident that they can release whenever they need to without complex coordination or late-stage testing. As with continuous integration, continuous delivery is a practice that requires a mixture of technical and organizational improvements to be effective.

# Why CI/CD ?

- Automate infrastructure creation and cleanup

  Reduce human error which could result in higher cost for unused infrastructure.

- Get to production faster

  By taking advantage of automated deployment, CI/CD pipeline can deliver our features to customers as soon as they are ready

- Automated Rollbacks

  In case the new version has any errors, automated rollback will take care of reverting to the last stable version as well as doing any necessary cleanup

# Why CI/CD

- Fail fast

  CI/CD adopts a fail-fast methodology, hence reducing running costs as much as possible

- Automated smoke tests

  To make sure the new release works as expected overall, with the option to rollback

- Automated alerts

  To alert dev team of any code breakage immediately at any step in the pipeline, hence saving time needed for investigation