

---

---

# CICD

Continuous Integration Continuous  
Deployment

---

# Overview

## **Continuous Integration**

The practice of merging all developers' working copies to a shared mainline several times a day. It's the process of "Making". Everything related to the code fits here, and it all culminates in the ultimate goal of CI: a high quality, deployable artifact!

## **Continuous Deployment**

A software engineering approach in which the value is delivered frequently through automated deployments. Everything related to dep



# Benefits of CICD

- **Automate Infrastructure**  
Automate Infrastructure Creation and clean up: Eliminating human errors and avoid unnecessary cost of unused or invalid infrastructure
- **Faster to production**  
By automating the pipeline to production this way we can deploy features as soon as created which will help increase revenue



# Benefits of CICD

## → Automated Rollback Triggered by Job Failure

Automate the process of rolling back and cleaning any infrastructure left which would help in reducing cost and lower down time

## → Catch Unit Test Failures

Unit tests are not neglected with CICD which will increase code quality and catch errors early before production which would decrease cost



# Benefits of CICD

## → Catch Compile Errors After Merge

Discover errors as soon as the developer make his commit which will help reduce the time of developers and reduce cost

## → 6 Automated Smoke Tests

Automate smoke test after deployment and automatic rollback in case of failure which will decrease downtime and reduce cost

---

**Thank You!**