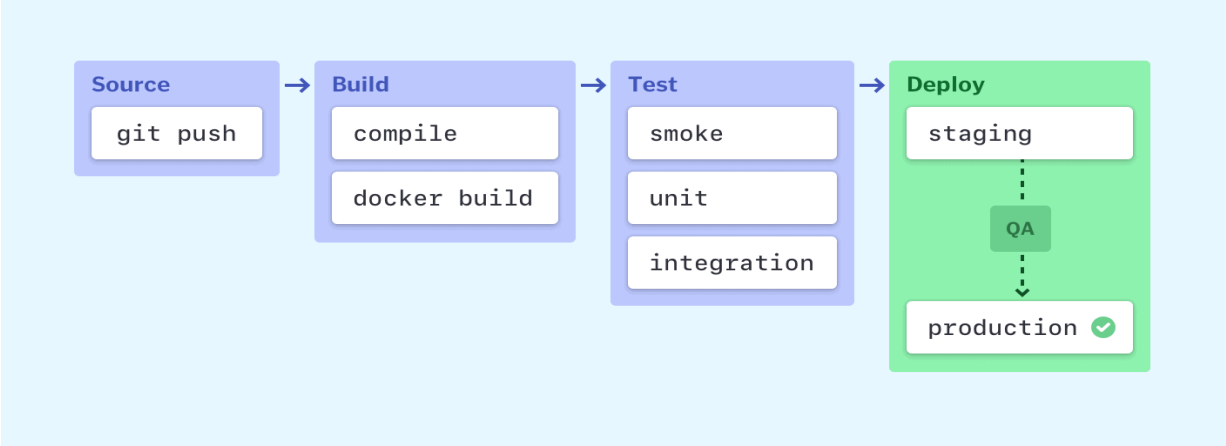
Fundamentals of CI/CD

A CI/CD pipeline automates your software delivery process. The pipeline builds codes, runs tests, and safely deploys a new version of the application (CD).

***Continuous integration (CI)** is practice that involves developers making small changes and checks to their code. Due to the scale of requirements and the number of steps involved, this process is automated to ensure that teams can build, test, and package their applications in a reliable and repeatable way. CI helps streamline code changes, thereby increasing time for developers to make changes and contribute to improved software.*

Some common CI-related phases might include:

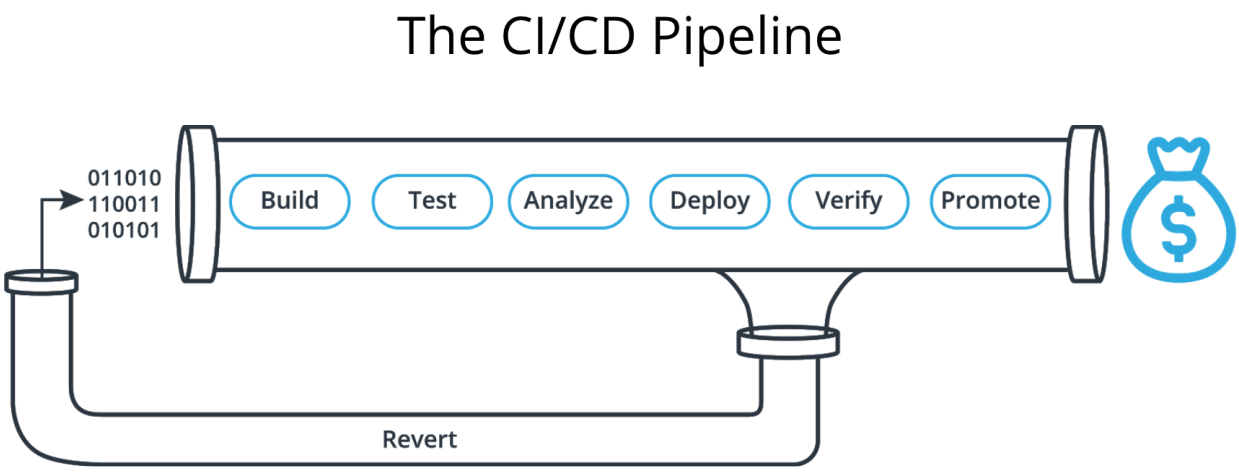
- *Compile*
- *Unit Test*
- *Static Analysis*
- *Dependency vulnerability testing*
- *Store artifact*



***Continuous Delivery** is the automated delivery of completed code to environments like testing and development. CD provides an automated and consistent way for code to be delivered to these environments.*

Continuous deployment *is the next step of continuous delivery. It is a software engineering approach in which the value is delivered frequently through automated deployments. Everything related to deploying the artifact fits here. It's the process of* ***"Moving"*** *the artifact from the shelf to the spotlight. Some common CD-related phases might include:*

- *Creating infrastructure*
- *Provisioning servers*
- *Copying files*
- *Promoting to production*
- *Smoke Testing (aka Verify)*
- *Rollbacks*



Benefits of including CI/CD pipeline in software application developments.

Technical Language	***Value***	***Translation***

Catch Compile Errors After Merge	*Reduce Cost*	*Less developer time on issues from new developer code*
Catch Unit Test Failures	*Avoid Cost*	*Less bugs in production and less time in testing*
Detect Security Vulnerabilities	*Avoid Cost*	*Prevent embarrassing or costly security holes*
Automate Infrastructure Creation	*Avoid Cost*	*Less human error, Faster deployments*
Automate Infrastructure Cleanup	*Reduce Cost*	*Less infrastructure costs from unused resources*
Faster and More Frequent Production Deployments	*Increase Revenue*	*New value-generating features released more quickly*
Deploy to Production Without Manual Checks	*Increase Revenue*	*Less time to market*

Automated Smoke Tests	***Protect Revenue***	***Reduced downtime from a deploy-related crash or major bug***
Automated Rollback Triggered by Job Failure	***Protect Revenue***	***Quick undo to return production to working state***

The above benefits clearly states that CI/CD help organisations to reduce cost of software application development and increase revenue.