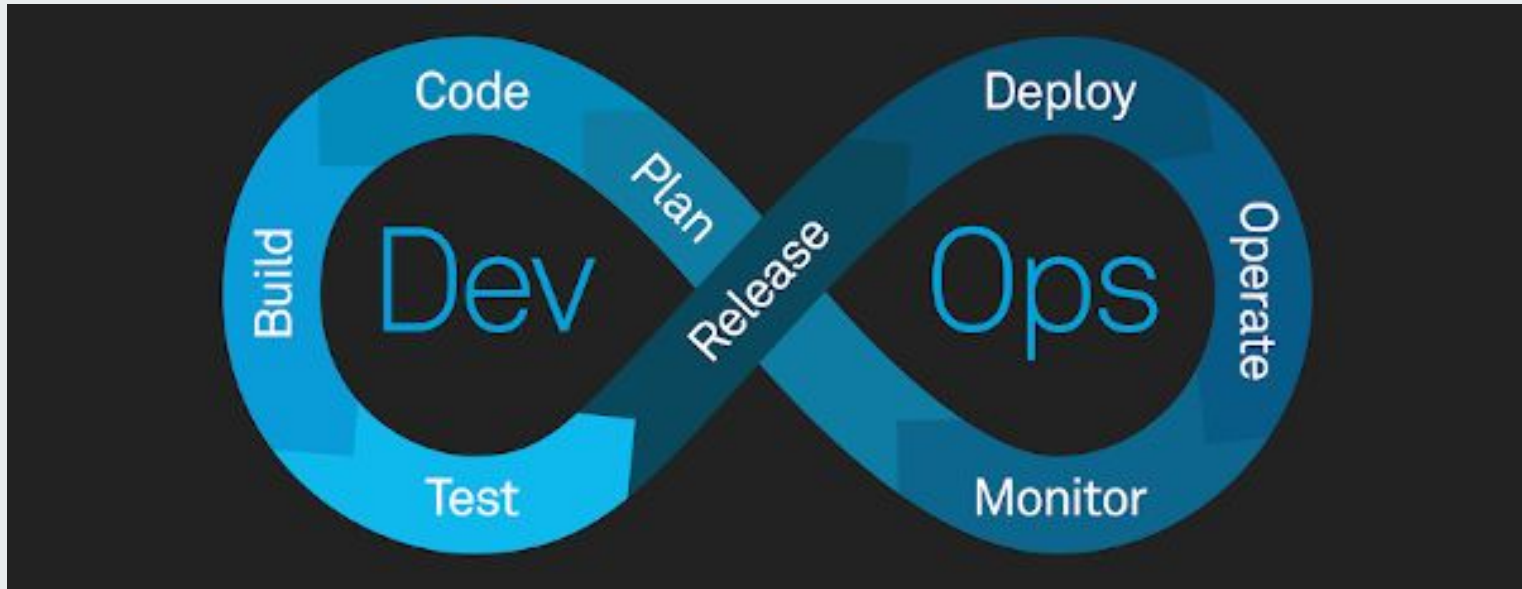


# CI/CD benefits

Presented by Mohamed Essam Ali Korany Kotb





# What is CI/CD

**Continuous integration** is all about the **source code**.

New changes to the code need to be validated, verified, exercised, worked over, massaged and squeezed to see if there are leaks. We do this by compiling, transpiling, linting, running unit tests, performing static analysis, checking dependencies for security vulnerabilities and other things.

**Continuous delivery/deployment** is all about **built code and deployment**.

Once the source code has been built in CI, we're ready to **ship it to servers** and devices either in the same network or elsewhere. Depending on your team's delivery process and deployment strategy, you might deploy to a staging or pre-production server for final testing or you might deploy to production right away. Before doing so, CD can run scripts to prepare the infrastructure, run smoke tests, and handle rollbacks and reverts if something doesn't go as planned.



# CI benefits

## What you gain:

- Less bugs get shipped to production as regressions are captured early by the automated tests. ⇒ faster productivity ⇒ more revenue
- Building the release is easy as all integration issues have been solved early. ⇒ faster productivity ⇒ more revenue
- Less context switching as developers are alerted as soon as they break the build and can work on fixing it before they move to another task. ⇒ less time wasted ⇒ faster productivity and higher revenue
- Testing costs are reduced drastically – your CI server can run hundreds of tests in the matter of seconds. (CI is triggered by a push) ⇒ less waste of resources ⇒ reduce cost
- Your QA team spends less time testing and can focus on significant improvements to the quality culture. ⇒ faster productivity and higher revenue



# CD benefits

## What you gain

- The complexity of deploying software has been taken away. Your team doesn't have to spend days preparing for a release anymore. ⇒ less time spent in release ⇒ faster productivity and so higher revenue
- You can release more often, thus accelerating the feedback loop with your customers. ⇒ customers are more satisfied ⇒ higher revenue
- There is much less pressure on decisions for small changes, hence encouraging iterating faster. ⇒ faster releases and so higher productivity and revenue
- You can develop faster as there's no need to pause development for releases. Deployments pipelines are triggered automatically for every change. ⇒ less time wasted ⇒ faster productivity and higher revenue



## CD benefits

- Releases are less risky and easier to fix in case of problem as you deploy small batches of changes. ⇒ find errors easier and less time consumed in releases ⇒ faster productivity and higher revenue
- Customers see a continuous stream of improvements, and quality increases every day, instead of every month, quarter or year. ⇒ customers are more happy with our services ⇒ more revenue since we can raise the contracts with them