What is CICD ?

Continuous integration and continuous deployment (CI/CD) is a set of ideas, processes, and capabilities that enables software changes to be delivered to users in a timely, repeatable, and secure manner by introducing automation into software development processes.

Continuous Integration & continuous Delivery:

- **CI:** An approach to be continually validating the state of a codebase through automated testing, it is best achieved through automated testing.

- **CD:** An approach to regularly deploying artifacts that successfully pass the CI phase to ensure confidence around the deployment.

Benefits of CICD

- **Higher efficiency of web development**: Increased productivity is one of the leading advantages of a CI/CD pipeline. You should automate your process if you have a review process that includes deploying code to development, testing, and production environments and entering multiple commands across several domains. This creates the need for a CI/CD framework.
- **Reduced risk of defects**: Finding and resolving defects late in the development process is costly and time-consuming. This is particularly true when problems arise with features already released to production.
- **Faster product delivery**: CD enables your team to provide customers with frequent and timely updates. When CI/CD is used, the entire team's efficiency increases, including the release of new features and fixes to problems. Businesses can address market shifts, security challenges, consumer needs, and financial pressures faster.
- **Exclusive log generation**: Logging information plays a vital role in observability. Logs provide a large volume of information to decipher what's happening beneath the UI and study program behavior. A CI/CD pipeline generates a lot of logging data at every level of the software development process.
- **Easier rollback of code changes**: One of the most exclusive benefits of a CI/CD pipeline is that it leads to the quick and easy rollback of code changes if there are any issues in the production environment after a release.