

# CI/CD

---

**A souped-up way to  
build and ship your  
products to market.**





# Why CI/CD?

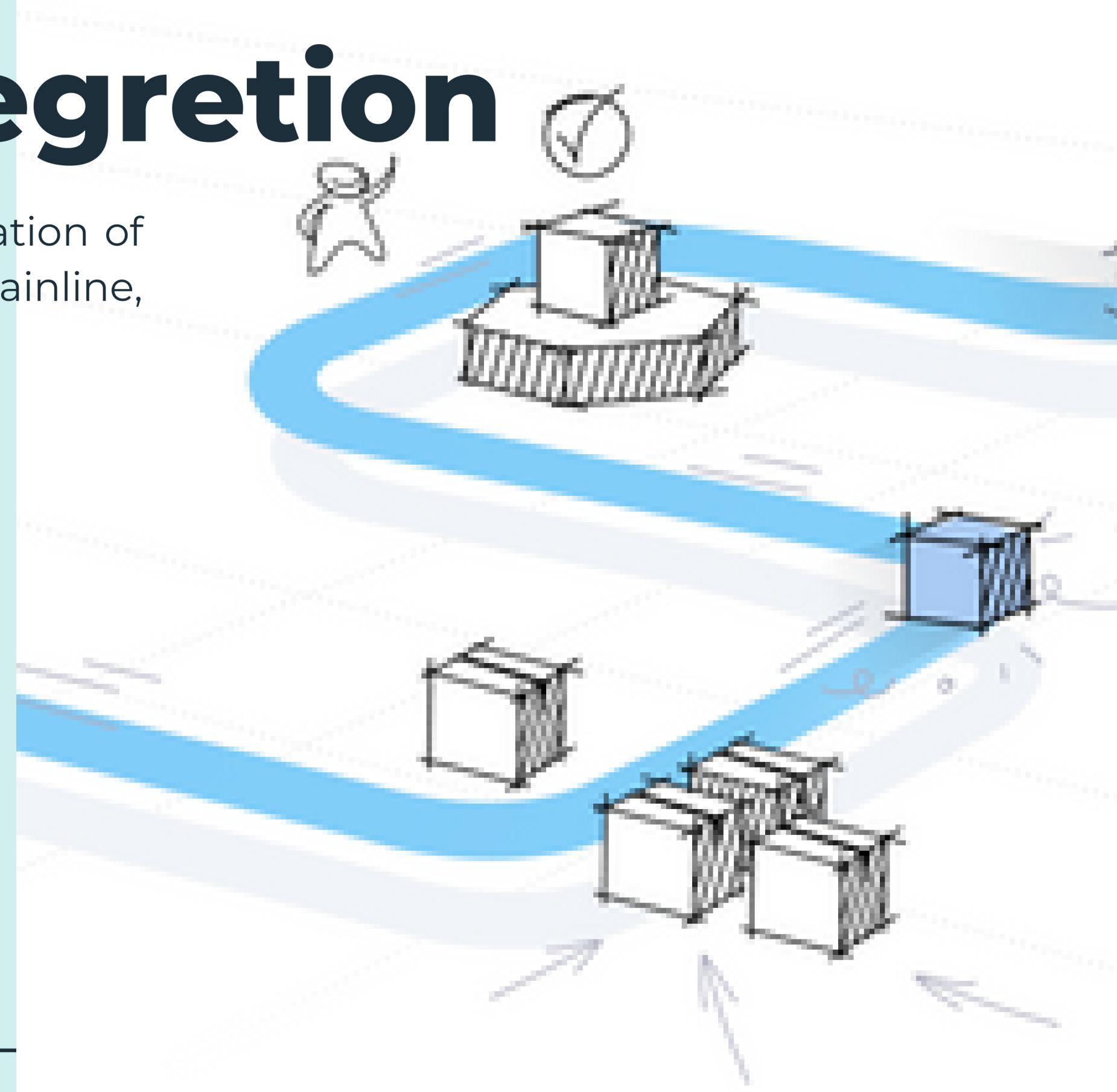
Implement automation for building and deploying our company's products.

**Continuous Integration - CI**  
**Continuous Delivery - CD**

Presented By : Jeffrey Kirigo

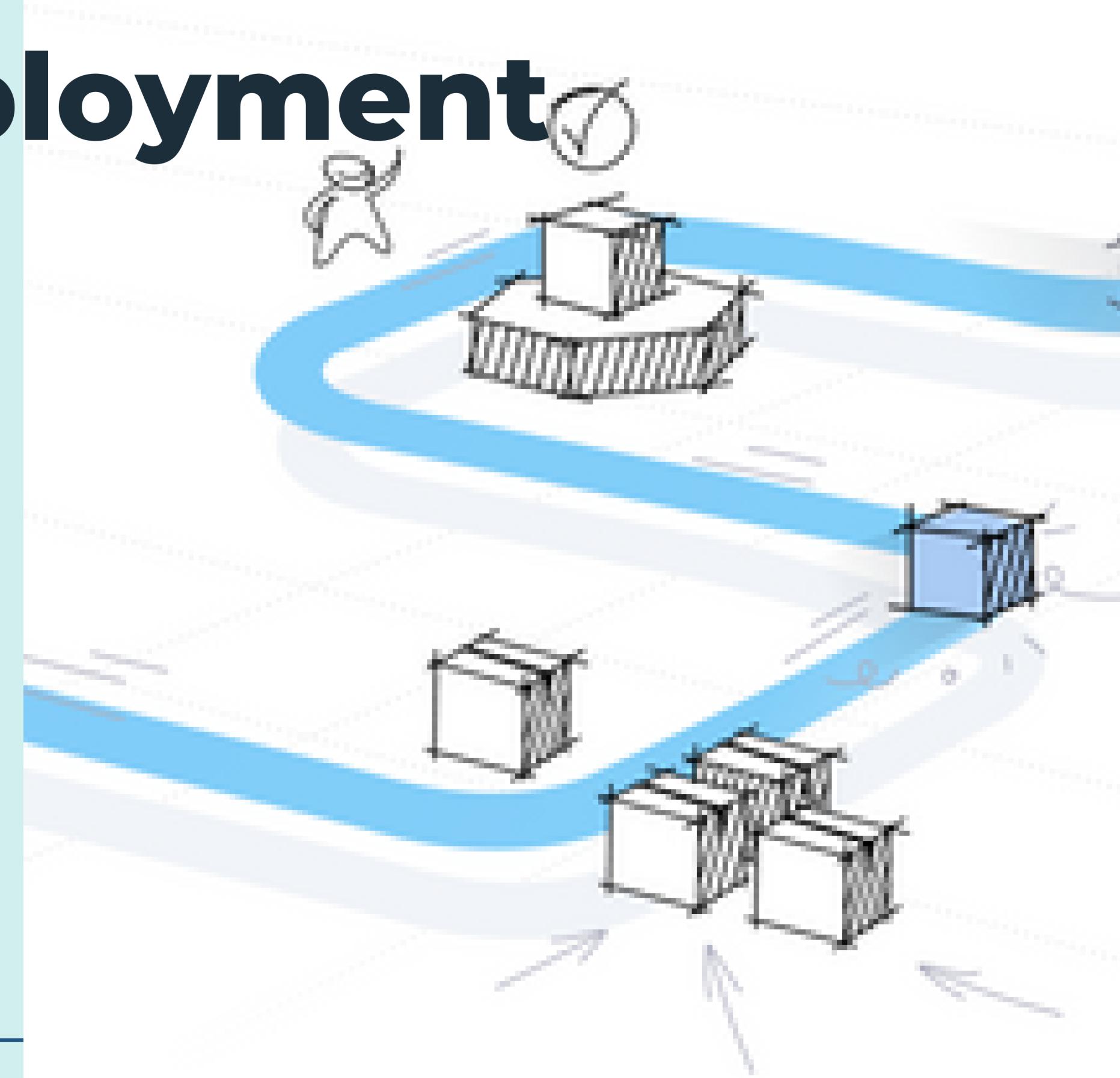
# Continuous Integration

- Is the practice of early and frequent integration of all developer code changes into a shared mainline, atleast several times a day.
- Everything related to the core code for the final product fits here.
- It entails the following critical phases: Building, testing, vulnerability check, creation of artifacts (Necessary file), static analysis
- Everything adds up to the ultimate goal of CI: a high-quality, deployable artifact!



# Continuous Deployment

- A software engineering practice in which value is frequently delivered through automated deployments.
- Everything here is related to moving the code-base(Application) from the 'Factory' to the 'shelf' or rather the spotlight.
- The CD-related phases include but are not limited to:
  - Creating and configuring infrastructure
  - Promoting to production
  - Verification - final test (Smoke test)
  - Rollbacks in case if any failure



# Benefits of CI/CD

The famous saying "a penny saved is a penny earned" - CI/CD axiom

## Reducing cost

Less human error, faster deployments  
Automate Infrastructure Creation

## Increase revenue

Faster and more frequent production  
deployments  
Deploy to Production Without Manual Checks

## Avoid costing

Detect Security Vulnerabilities early enough -  
avoid costly security holes

## Protecting revenue

reducing downtime from a deploy-related crash  
or a major bug - smoke tests

# Get in Touch



**Contact us to get more info**

-  **jeffreykirigo75@gmail.com**
-  **Nairobi, Kenya**
-  **jey-codes.tech**