

## **Explain the Fundamentals and Benefits of CI/CD to Achieve, Build, and Deploy Automation for Cloud-Based Software Products as presented by Ndimneze Chukwudi Stanley, in pursuit of completion of ALX/udacity devops course.**

### **A. Improves code quality and developers confidence.**

Implementing continuous integration, developers do frequent code commits which makes it easier to identify defects and quality related issues. On the other hand, following the conventional methods force developers to do the same task again and again. This ultimately affects their efficiency and brings less return on investment. Continuous integration leads to lesser merge conflicts due to the shorter commit cycles which ultimately save a lot of business hours which can be spent on creating more new interesting features.

### **B. Saves money for the business when time to delivery is accounted for.**

CI/CD reduces the MTTR because the code changes are smaller and fault isolations are easier to detect. One of the most important business risk assurances is to keep failures to a minimum and quickly recover from any failures that do happen. Application monitoring tools are a great way to find and fix failures while also logging the problems to notice trends faster.

### **C. Saves money for the business when time to delivery is accounted for.**

CI/CD continuously merges codes and continuously deploys them to production after thorough testing, keeping the code in a release-ready state. It's important to have as part of deployment a production environment set up that closely mimics that which end-users will ultimately be using. Containerization is a great method to test the code in a production environment to test only the area that will be affected by the release.