Continuous Integration Training Course Online

Curriculum 3 Days

Day - 1

- **Continuous Integration Concept and Process**
  - Continuous Integration and Delivery Process
  - Problem That DevOps Solves
  - Explore the DevOps background, approach, and best practices
  - Integrate test automation with DevOps
  - Implement continuous testing
  - Learn how DevOps practices and principles improve software quality and efficiency
  - Understand the differences between Continuous Integration and traditional operational methodologies
  - Discover the major steps required to successfully implement delivery pipelines
  - Continuous Integration Transition to a Project
  - Understanding the Continuous Integration and Deployment (CI/CD)
  - Implement DevOps - Organization & Culture
- **Tools and Technologies Introduction to Continuous Integration**
  - What Is CI?
  - Why CI?
  - CI Philosophy
  - Advantages of CI Within Software Development
  - CI as an Enabler
  - Benefits of Continuous Integration
- **How to Implement CI**
  - Setting Up a CI Environment
  - CI Tools
  - Automated Builds
  - Automated Tests
  - Committing Code
  - Creating a Full-featured CI Environment
  - Components of a Full-featured Environment
  - Requirements of a CI Environment
  - Organizational Impact and Buy-in
  - Developer Philosophy
- **DB Integration**
  - Automated DB Integration
  - Continuous DB Integration
- **Continuous Testing**
  - Automated Unit, Integration, System and Functional Tests
  - Writing Tests for Defects
  - Integration of Automated Testing With CI

Day - 2

- **Continuous Quality**
  - Code Inspection vs. Code Testing
  - Automated Code Analysis
  - Code Analysis Tools
- **Continuous Deployment**
  - Philosophy
  - Release and Labeling Strategies
- **Continuous Improvement**
  - The Feedback Loop
  - The Metrics Loop
  - The Improvement Cycle

- **Best Practices**
  - Best of Breed Tools
  - Best Practices for Quality Software
  - Best Practices for CI
  - Rolling Out CI Within Your Organization
  - Defining and Measuring Metrics and Reporting
  - Defining Proper Metrics
  - Measuring and Reporting
  - Implementing a Change Strategy to Achieve Desired Metrics
- **Minimum Requirements**
  - Overview
  - The Check-In Dance
  - Continuous Integration Do's and Don'ts
  - Summary

Day - 3

- **Building a Solution**
  - Overview
  - Hello, Continuous Integration
  - Recommended Solution Layout
  - The Software
  - Building on the CI Server
  - Build Failure Notifications
  - Receiving a Build Failure Notification
  - Summary
- **Build Scripts**
  - Build Scripts
  - Overview
  - Why Do We Need a Build Script?
  - Decision Point: How to Add Build Steps
  - Recommendation: Do Not Modify csproj/vbproj
  - Recommendation: Do Not Script csc.exe/vbc.exe
  - Recommendation: Script Everything, but Compile with MS-Build
  - Why the Command Line?
  - Hello, MS-Build
  - Defining Targets
  - Running from the Command Line
  - Compiling the Solution
- **Integrating External Tools**
  - Integrating External Tools
  - Overview
  - External Tools
  - Test Runners
  - Integrating N Unit via Exec Task
  - Failing the Build with Exit Codes
  - Running Tests on the Build Server
  - Displaying Test Reports
  - Coverage
  - 100% Coverage Myth
  - Integrating Coverage
  - Integrating N Cover via Custom MSBuild Tasks
  - Failing the Build on Low Coverage
  - Running Coverage on the Build Server
  - Static Code Analysis
  - Integrating FxCop
  - Failing the Build by Analyzing XML Output
  - Summary
- **Deployment Automation**
  - Overview
  - Packaging/Deployment Options