

**Integration Testing with Docker and Testcontainers**

**Course Number:** DVOP-154  
**Duration:** 0.5 days

**Overview**

This live, interactive Integration Testing with Docker and Testcontainers training teaches attendees how to set up and run integration and functional tests with open-source library Testcontainers. Attendees learn how to stand up lightweight, disposable Docker instances that run applications as reliable test fixtures.

**Prerequisites**

* A basic understanding of Docker and the Java programming language
* Hands-on experience with common Java frameworks and dependencies
* Experience with writing test code with the test framework JUnit 4 or JUnit 5

**Materials**

All attendees receive a copy of the instructor’s presentation and related code.

**Software Needed on Each Student PC**

* A computer with Java 11 OpenJDK and Docker Engine installed
* A GitHub account
* A Docker Hub account

**Objectives**

* Understand the purpose, differences, and trade-offs between test types
* Use Testcontainers to write integration tests
* Write integration tests for different scenarios
* Automate test execution in Maven and Gradle builds

**Outline**

* Problems with Integration Testing and How Testcontainers Can Help
  + The testing pyramid
  + What are Testcontainers?
  + Testcontainers technical quickstart
  + Your typical problems with integration testing
  + Using Testcontainers for a Java-based project with JUnit 5
* Implementing Typical Integration Test Scenarios
  + Testing database services
  + Testing multiple services together with Docker Compose
  + Building a container on the fly
  + Using a database module
  + The Docker Compose module
  + The generic container module
* Going Further
  + Comparing Testcontainers with JUnit 4, JUnit 5, and Spock
  + Using Testcontainers with different Java frameworks
  + Testcontainers on continuous integration
  + Testcontainers for languages other than Java
  + Using Testcontainers on GitHub Actions
* Conclusion