

## Preparation Guide for the Coding Exercises

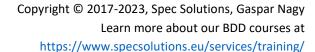
Automation with Cucumber Java Tutorial

## MACHINE PREREQUISITES

- Approximately 500MB free space is necessary in total (including size of dependencies).
- Java 8 JDK (or later) with Maven installed
- Optional: Java IDE (e.g. IntelliJ, Eclipse or Visual Studio Code)

## PREPARATION STEPS

- Clone the coding exercises from GitHub (<a href="https://github.com/specsolutions/20230530-ExpoQA-AutomationWithCucumberJava">https://github.com/specsolutions/20230530-ExpoQA-AutomationWithCucumberJava</a>) or <a href="download it as zip">download it as zip</a> and unzip it. <a href="Avoid using shared drives!">Avoid using shared drives!</a> We recommend using a folder with a shorter path to avoid reaching the maximum folder path limit of Windows.
- Now we are going to test your setup. For that we will use the B1 exercise as an example. Please perform the following steps carefully. If you find any problems, please inform us immediately at <a href="mailto:courses@specsolutions.eu">courses@specsolutions.eu</a>, so that we can fix the problems in time.
  - We will refer the mvn command a few times below. If that does not work, you can also use the mvnw command instead (mvnw is an mvn wrapper that chooses the right version of mvn).
- Download packages: Download packages for the B1 exercise (this will ensure that all packages that are used by the exercises are downloaded to the machine)
  - On a command line shell, go to the B1 folder in the downloaded repository.
  - o Invoke: mvn compile
- Test compilation and test execution: Run tests of B1 exercise: there should be one passing test.
  - On a command line shell, go to the B1 folder again.
  - o Invoke: mvn test
  - The command should complete without errors and report "Tests run: 1, Failures: 0, Errors: 0,
    Skipped: 0"
  - o Invoke: mvn spring-boot:run
  - The command should complete without errors and you should able to navigate to <a href="http://localhost:8019">http://localhost:8019</a> and see a pizza website, with some pizzas on the menu.
- Test IDE setup: Open B1 exercise in your IDE and run the demo app and the tests.
  - Open/import pom.xml from the B1 exercise in your IDE.
  - Build solution. Make sure it builds successfully.





000-

- Run tests (e.g. CucumberApiTest class) from IDE. You should get the same result as in Maven.
- Open home.feature file. You should see a feature file with syntax. If not: you need a Cucumber plugin.
- o Run the sample application. You should able to navigate to <a href="http://localhost:8019">http://localhost:8019</a>.

CONGRATULATIONS, YOU ARE READY TO GO!