

#### eine... 🍥 3 🕓 À l'instant

# karate apache 1.4.1 not available dependency dans pom

### :号 Sources



# 常 Réponse

The issue you are facing is related to the dependency version of karate-core or karate-junit5 in your Maven project's pom.xml file. Based on the search results:

The latest stable version of Karate at the time of this query (June 10, 2024) is 1.4.1. <sup>3</sup> However, your pom.xml is trying to use version 0.9.6 of the karate-apache artifact, which is an older version. <sup>4</sup>

To resolve this, you need to update the version in your pom.xml to the latest 1.4.1 version:

Alternatively, you can use the karate-core artifact with the 'all'



classifier to avoid dependency conflicts: 4



Make sure to also update the `<karate.version>` property in your pom.xml to 1.4.1. 1

This should resolve the dependency issue you are facing. 1 3

# run compose with karate framework



⇔ Partager





Start Creating Karate Framework from Scratch, Add Feature File, Create Test Runner, Execute and Verify Reports

Learn Automated API Testing using Karate Framework from Scratch – Session 2





To run tests in parallel using the Karate fr leverage the `parallel()` method provic class. Here's how you can se up:

## O Politique en matière de cookies

Nous utilisons des cookies pour améliorer votre expérience. En cliquant sur "Accepter tous les cookies" ou en sélectionnant "Cookies nécessaires", vous acceptez notre <u>politique de confidentialité</u>.

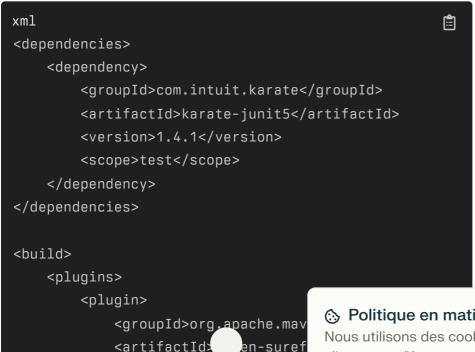
Accepter tous les cookies

`ParallelRunner.java`:

```
java
                                                    import com.intuit.karate.Results;
import com.intuit.karate.Runner;
import org.junit.Test;
import static org.junit.Assert.assertTrue;
public class ParallelRunner {
    @Test
    public void testParallel() {
        Results results = Runner.path("classpath:path/
                                .tags("~@ignore") // /
                                .parallel(5); // run t
        assertTrue(results.getErrorMessages(), results
}
```

#### In the above example:

- `Runner.path("classpath:path/to/features")` specifies the location of your feature files.
- `tags("~@ignore")` runs all tests except those tagged with `@ignore`.
- `parallel(5)` runs the tests in 5 parallel threads.
- 2. In your `pom.xml`, add the Karate dependencies and configure the `maven-surefire-plugin` to run the `ParallelRunner` class: 1



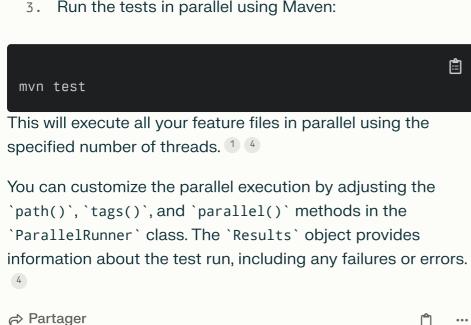
## Politique en matière de cookies

Nous utilisons des cookies pour améliorer votre expérience. En cliquant sur "Accepter tous les cookies" ou en sélectionnant "Cookies nécessaires", vous acceptez notre politique de confidentialité.

Accepter tous les cookies

```
</configuration>
        </plugin>
    </plugins>
</build>
```

Run the tests in parallel using Maven:



# run a compose file docker with karate before start test

# **常 Réponse**

To run a Docker Compose file before starting your Karate tests, you can use a setup script or a Maven plugin. Here's an example of how you can achieve this using the `dockercompose-maven-plugin`:

1. Add the `docker-compose-maven-plugin` to your `pom.xml` file:



```
<execution>
                     <id>up</id>
                     <phase>pre-integration-test</phase</pre>
                     <goals>
                         <goal>up</goal>
                     </goals>
                     <configuration>
                         <composeFile>${project.basedir
                     </configuration>
                 </execution>
                 <execution>
                     <id>down</id>
                     <phase>post-integration-test</phas</pre>
                     <goals>
                         <goal>down</goal>
                     </goals>
                     <configuration>
                         <composeFile>${project.basedir
                     </configuration>
                 </execution>
            </executions>
        </plugin>
    </plugins>
</build>
```

- Replace `\${project.basedir}/path/to/dockercompose.yml` with the actual path to your Docker Compose file.
- 3. In the `<execution>` sections, you can see that the `up` goal is bound to the `pre-integration-test` phase, which means it will run before your Karate tests. The `down` goal is bound to the `post-integration-test` phase, which means it will run after your Karate tests have completed.
- 4. Add the Karate dependencies and configure the `maven-surefire-plugin` to run your Karate tests:

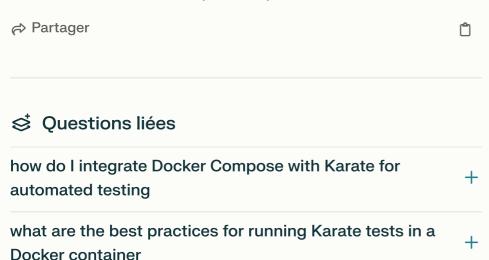


5. Run your Karate tests using Maven:



This will first start the Docker Compose services (`docker-compose up`), then run your Karate tests, and finally stop the Docker Compose services (`docker-compose down`).

Make sure to replace the plugin versions and configurations with the appropriate values for your project. Additionally, ensure that your Docker Compose file is correctly configured and accessible from the specified path.



how can Loonfigura Docker Company to

# O Politique en matière de cookies

Nous utilisons des cookies pour améliorer votre expérience. En cliquant sur "Accepter tous les cookies" ou en sélectionnant "Cookies nécessaires", vous acceptez notre <u>politique de confidentialité</u>.

Accepter tous les cookies