Karate Runner Class – All Scenarios (Q&A; Interview Format)

Scenario 1: Run ALL feature files from a folder
Q: What will this runner execute?
A: It will execute all *.feature* files present under:
src/test/java/scripts/UdemyKarate

Scenario 2: Folder path is wrong
Q: What happens if the path is incorrect?
A: Execution fails with:
No features found

Scenario 3: Only ONE feature file present
Q: What if the folder has only one feature file?
A: That single feature file will be executed.

Scenario 4: Folder is empty
Q: What if there are no .feature files in the folder?
A: Runner fails with No features found error.

Scenario 5: Why classpath: is used?
Q: Why not use absolute path?
A: classpath makes execution portable across machines and CI pipelines.

Scenario 6: Remove relativeTo(getClass())
Q: Will it still work?
A: Sometimes yes, but not reliable. relativeTo(getClass()) ensures correct path resolution.

Scenario 7: Method name changed
Q: Can we rename testSample()?
A: Yes. Method name does not matter. JUnit identifies it using @Karate.Test.

Scenario 8: Remove @Karate.Test
Q: What happens if annotation is removed?
A: JUnit will not execute this method. Runner will be ignored.

Scenario 9: Runner class location changed
Q: What if runner is outside src/test/java?
A: Maven Surefire will not detect it, tests won't run.

Scenario 10: Running via Maven
Q: How is this runner executed in CI/CD?
A: By running *mvn test*. Maven automatically triggers this runner.

Scenario 11: Multiple runner classes
Q: Can we have more than one runner class?
A: Yes. Common practice includes SmokeTestRunner and RegressionTestRunner.

Scenario 12: Parallel execution needed
Q: How will you modify this runner?
A:
Karate.run("classpath:/scripts/UdemyKarate").parallel(5);

Scenario 13: Run by tag
Q: How to run only @smoke tests using this runner?
A:
Karate.run().tags("@smoke").relativeTo(getClass());

Scenario 14: Can this runner execute UI tests?
A: Yes. Same runner can execute API, UI, and performance Karate tests.

Most Important Interview One-Liner
This is a Karate JUnit5 runner class that executes feature files from the classpath, integrates with Maven and CI, and supports scalable execution through tags and parallelism.